BAYESIAN HIERARCHICAL MODEL FOR COMBINING TWO-RESOLUTION METROLOGY DATA

A Dissertation

by

HAIFENG XIA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2008

Major Subject: Industrial Engineering

BAYESIAN HIERARCHICAL MODEL FOR COMBINING TWO-RESOLUTION METROLOGY DATA

A Dissertation

by

HAIFENG XIA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Yu Ding
Committee Members,	Bani K. Mallick
	Brett A. Peters
	Jyhwen Wang
Head of Department,	Brett A. Peters

December 2008

Major Subject: Industrial Engineering

ABSTRACT

Bayesian Hierarchical Model for Combining Two-resolution Metrology Data.

(December 2008)

Haifeng Xia, B.S., Tianjin University, Tianjin, China;

M.S., Tianjin University, Tianjin, China

Chair of Advisory Committee: Dr. Yu Ding

This dissertation presents a Bayesian hierarchical model to combine two-resolution metrology data for inspecting the geometric quality of manufactured parts. The high-resolution data points are scarce, and thus scatter over the surface being measured, while the low-resolution data are pervasive, but less accurate or less precise. Combining the two datasets could supposedly make a better prediction of the geometric surface of a manufactured part than using a single dataset. One challenge in combining the metrology datasets is the misalignment which exists between the low- and high-resolution data points.

This dissertation attempts to provide a Bayesian hierarchical model that can handle such misaligned datasets, and includes the following components: (a) a Gaussian process for modeling metrology data at the low-resolution level; (b) a heuristic matching and alignment method that produces a pool of candidate matches and transformations between the two datasets; (c) a linkage model, conditioned on a given match and its associated transformation, that connects a high-resolution data point to a set of low-resolution data points in its neighborhood and makes a combined prediction; and finally (d) Bayesian model averaging of the predictive models in (c) over the pool of candidate matches found in (b). This Bayesian model averaging procedure assigns weights to different matches according to how much they support the observed data, and then produces the final combined prediction of the surface

based on the data of both resolutions.

The proposed method improves upon the methods of using a single dataset as well as a combined prediction without addressing the misalignment problem. This dissertation demonstrates the improvements over alternative methods using both simulated data and the datasets from a milled sine-wave part, measured by two coordinate measuring machines of different resolutions, respectively.

To my parents

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor, Professor Yu Ding, for his mentorship, guidance and support throughout my Ph.D. study at Texas A&M University, as well as the development of this dissertation. I can not thank him enough for his help in every aspect of my Ph.D. study.

I am also thankful to Professor Bani K. Mallick and Professor Jyhwen Wang, who served as my Ph.D. committee members and my coauthors on technical papers. I really appreciate Professor Mallick's guidance on Bayesian methodologies and Professor Wang's support on manufacturing related issues.

I would also like to extend my gratitude to my Ph.D. committee member, Professor Brett A. Peters, for his constructive suggestions and invaluable discussions throughout the development of my Ph.D. research and his careful review of this dissertation.

I also owe many thanks to Ms. Hui Lin, Mr. Ahbishek K. Shrivastava and other members of the Advanced Metrology Lab for their friendship and support during my Ph.D. study. My time at Texas A&M University would have never been as pleasant and enjoyable without their friendship.

Last but not least, I am deeply grateful to my parents, my brother and Mr. Christopher T. Rodenbeck for their love, encouragement and enormous support.

NOMENCLATURE

$\mathbf{x}_i$	i-th input to a measuring device, nominal position to be measured
$\{\mathbf{x}_i\}_{i=1:m}$	Set of locations where measurements are made
$\{\mathbf{n}_i\}_{i=1:N}$	Set of locations where measurements are to be predicted
$\mathbf{X}$	Location matrix where measurements are made
$\mathbf{X}_l$	Location matrix where low-resolution measurements are made
$\mathbf{X}_h$	Location matrix where high-resolution measurements are made
$\mathbf{a}_i$	i-th output from a measuring device, coordinate measurement of a point on a manufactured surface
$\mathbf{a}_j^l$	j-th low-resolution measurement
$\mathbf{a}_i^h$	i-th high-resolution measurement
D_l	Set of low-resolution measurements
D_h	Set of high-resolution measurements
m_l	Total number of low-resolution measurements
m_h	Total number of high-resolution measurements
$\mathbf{p}_i$	Approaching direction used by a measuring device to measure $\mathbf{x}_i$
y_i	i-th response, a projected value of $\mathbf{a}_i$ to its associated $\mathbf{p}_i$: $y_i = \mathbf{a}_i^T \mathbf{p}_i$
y_l	Low-resolution response
y_h	High-resolution response
$\boldsymbol{\varphi}$	Rigid body transformation parameters
ρ_u, ρ_v, ρ_w	Rotation around the u-, v- and w-axis, respectively
t_u, t_v, t_w	Translation along the u-, v- and w-axis, respectively
$\boldsymbol{\psi}$	Dimensional parameters for a geometry
$\boldsymbol{\psi}^*$	Nominal/design values for dimensional parameters of a geometry

$\boldsymbol{\beta}$	Parameters for the mean component in the single-resolution data model, and $\boldsymbol{\beta} = (\boldsymbol{\varphi}, \boldsymbol{\psi})$
$\boldsymbol{\beta}_l$	Mean parameters for the low-resolution data model
$\xi(\cdot)$	Systematic error that is assumed to a Gaussian process of zero mean and a certain covariance
κ^2	Variance of systematic error $\xi(\cdot)$
κ_l^2	Variance of low-resolution systematic error
R	Correlation function
$\mathbf{R}$	Correlation matrix
$\boldsymbol{\nu}$	Scale parameter for the correlation function
ϵ	Random error at single-resolution level
ϵ_l	Random error at low-resolution level
σ_ϵ^2	Variance of random error
σ_l^2	Variance of low-resolution random error
$\boldsymbol{\Sigma}$	Covariance matrix
$\mathbf{I}$	Identity matrix
η_l	Filtered low-resolution response, free of low-resolution measurement noises
$M(\mathbf{0}, \boldsymbol{\psi}^*)$	Nominal geometry
$M(\boldsymbol{\varphi}, \boldsymbol{\psi})$	Dashed-line geometry, including both dimensional and location variations
$\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$	Point on dashed-line geometry that corresponds to $\mathbf{x}_i$
$g(\mathbf{x}_i, \boldsymbol{\beta})$	Projected value of $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ onto the associated approaching direction $\mathbf{p}_i$: $g(\mathbf{x}_i, \boldsymbol{\beta}) = \mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})\mathbf{p}_i$
h^*	True form error
$\hat{h}_{\mathrm{OLS}}(m)$	Estimated form error using the OLS method and m data points
$\hat{h}_{\mathrm{MZ}}(m)$	Estimated form error using the MZ method and m data points

$\hat{h}_{\mathrm{GP}}(m)$	Estimated form error using the GP method and m data points
$\mathbf{M}$	Matching matrix that records the matching relationship between low-resolution and high-resolution datasets
$\mathbf{M}_k$	k-th consistent match found by the heuristic matching algorithm
K	Total number of consistent matches found
ϖ	Distance threshold used in the heuristic matching algorithm
$\mathbf{H}$	Rigid body rotation matrix
$\boldsymbol{\alpha} = (\alpha_1, \alpha_0)$	Scale and location coefficient in the neighborhood linkage model
e	Residual in the neighborhood linkage model
$K(\mathbf{x}_i, \mathbf{x}_j)$	Kernel function which defines the neighborhoods of high-resolution data points
$\boldsymbol{\lambda}$	Kernel widths for all coordinate axes
λ_0	Upper bound for the prior distributions of kernel widths
$\Psi(\cdot)$	Modified tri-cube kernel function
$\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x}_0)$	Row vector defined as $(\sum_{i=1,\ldots,m_l} K(\mathbf{x}_0, \mathbf{x}_i)\eta_l(\mathbf{x}_i), 1)$
$\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x}_0)$	Abbreviation of $\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{X}_h)$ whose j-th row is defined as $(\sum_{i=1,\ldots,m_l} K(\mathbf{x}_j, \mathbf{x}_i)\eta_l(\mathbf{x}_i), 1)$
$\boldsymbol{\theta}$	All the parameters involved in the single-resolution data model
$\boldsymbol{\theta}_l$	All the parameters involved in the low-resolution data model
$\boldsymbol{\theta}_h$	All the parameters involved in the neighborhood linkage model

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

CHAPTER I

INTRODUCTION

A. Motivation

This dissertation attempts to address the problem of combining data from two different resolutions to improve both efficiency and measuring resolution such that the inspection of geometric features on machined parts for compliance with tolerance specifications will also improve.

Functional requirements or assembly conditions are normally translated into geometric constraints on parts. In other words, engineers often design parts to have a certain geometric form and size to achieve certain functions. However, manufacturing operations are hardly perfect, so manufactured features inevitably deviate from their nominal designs. The deviation is known as manufacturing error. Therefore, designers assign tolerances to specify the allowable range of manufacturing errors. If the tolerance controls the errors associated with a dimension of a part, e.g., the radius of a shaft, it is known as dimensional tolerance; if the tolerance controls the errors associated with a geometric form, e.g., the straightness or roundness features shown in Fig. 1, it is known as geometric tolerance or form tolerance. Between dimensional features and geometric features, the latter are more difficult to assess and to control, but their integrity is closely related to good quality and proper functionality. Please refer to the example of an automotive transmission in [60].

In order to assure dimensional quality of machined products, a crucial step is to take metrology data (i.e., coordinate measurements) of the geometric features and then check their compliance with tolerance specifications.

The journal model is *IEEE Transactions on Automatic Control.*

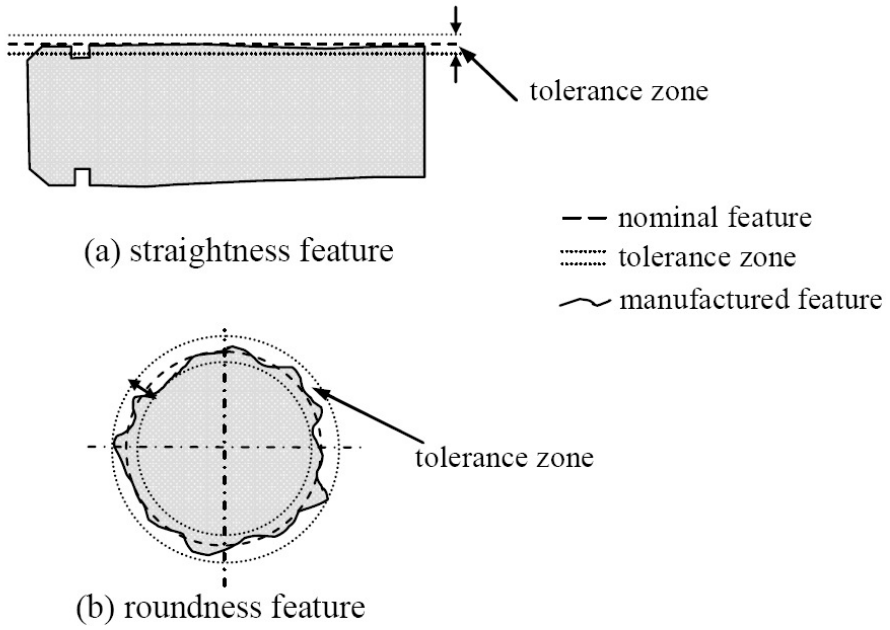

Fig. 1. Illustration of geometric features and their form tolerance zones

Traditionally, a Coordinate Measuring Machine (CMM) with a mechanical touch probe is used due to its accuracy and versatility in measuring complicated geometries. Lately, a CMM with an optical/laser sensor probe (hereinafter OCMM) has been introduced into industry practice as a complement, sometimes a replacement, of the CMM. An OCMM takes measurements by forming an image consisting of the laser light reflected from the part surface. Fig. 2 shows both a CMM and an OCMM.

The resolution of an OCMM is typically much lower than that of a CMM. By resolution, we refer to the smallest spatial distance that a measuring device can distinguish. A high-resolution device can distinguish two closely positioned points and pick up fine spatial features on a product surface. It therefore attains a higher accuracy (i.e., a smaller bias) and a better precision (i.e., a smaller variability) in its measurements than its low-resolution counterpart. According to [49], a CMM could have a resolution as fine as 0.5 microns, while an OCMM typically has a resolution

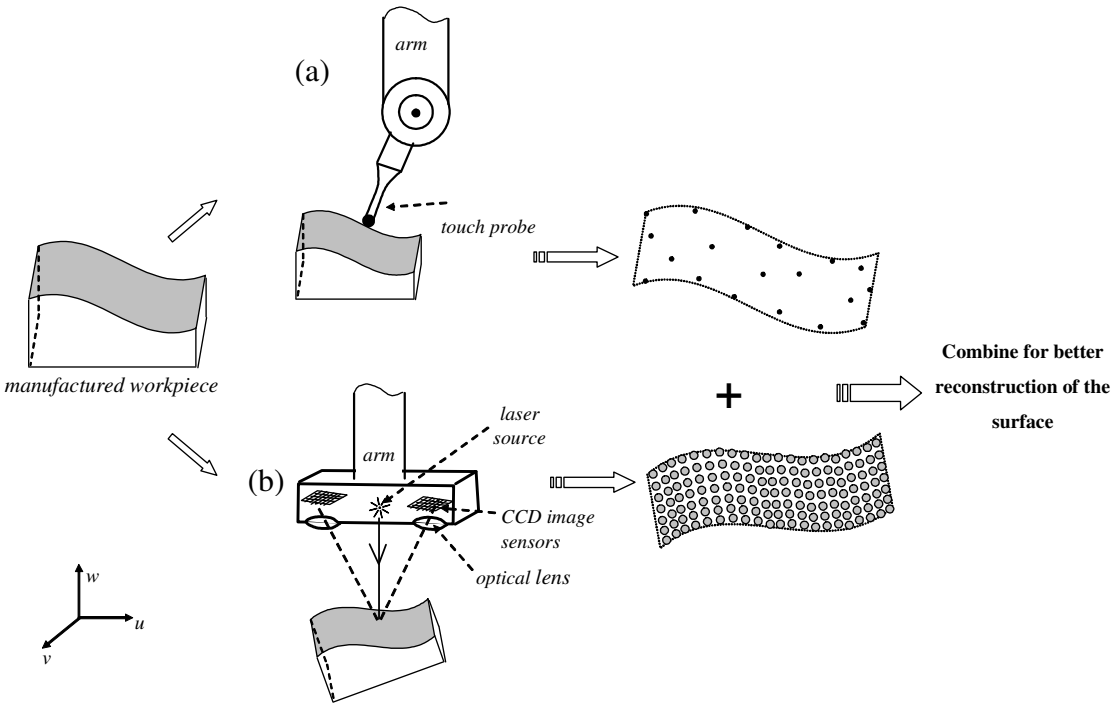

Fig. 2. Two-resolution data from CMM with touch probe and laser scanner

on the order of 10 microns. Even though the high-resolution CMM is capable of measuring surfaces at much finer scales, doing so is very time-consuming. Common practice is to take measurements of a few locations scattered over the product surface. The OCMM can scan the entire surface of a product much faster than a mechanical CMM, but each measurement is of a lower resolution (i.e., associated with high degree of inaccuracy and uncertainty). As such, when both metrology devices are used, one would have a large set of low-resolution data and a much smaller set of high-resolution data. The scattered points (dark) and the densely arranged points (grey) in Fig. 2 illustrate the two-resolution metrology data.

The datasets of different resolutions complement each other in the information needed to reconstruct the surface, which is usually done by predicting the coordinates

Table I. Characteristics of high- and low-resolution metrology data

	Advantages	Disadvantages
High-resolution data	High accuracy and high precision	More costly to obtain. Very scarce and fail to capture the overall trend of a product surface
Low-resolution data	Easier to obtain and usually densely cover the entire product surface. Therefore, capture the overall trend of a product surface well	Low accuracy and low precision

of unmeasured locations on the same part. Table I summarizes the characteristics of high-resolution and low-resolution data. An integrated analysis should be able to produce better results, combining strengths across multiple sources. Our research is to devise a method taking full advantage of both datasets and making good predictions of the product surface.

A unique challenge in combining data for the metrology application is the misalignment existing between the data points of different resolutions. The misalignment happens because when using two metrology devices to measure the same part, it is often the case that the data are not collected with respect to a common coordinate system. Consequently, the mismatch of the coordinate systems makes it difficult to decide how the data points in the two datasets correspond to one another. For example, two spatial points of the same (1.0, 1.0, 1.0) measurements in the two different datasets could correspond to two completely different points on the actual part. Even

if the part is measured on a single measuring platform carrying both a touch and a laser probe, independent calibration is needed for each probe, and the coordinates of measured points is contaminated by noises, still resulting in misalignments between the datasets.

B. Research Objective

The objective of this dissertation is to develop a methodology that can combine the misaligned multi-resolution metrology data effectively to better predict the product surface. Such a successful methodology can directly contribute to improving measurement resolutions and efficiency such that engineers can do a geometric inspection faster and more accurately.

We plan to combine metrology data from different resolutions in two steps. The first step is to build a single-resolution data model using a Gaussian process model and decompose single-resolution data into three components: designed geometric form, systematic deviation from the designed geometry, and random errors. The purpose is to link the data at different locations to achieve better information about the product surface. Ideally, the proposed method should be a general model that works for various geometries. The second step is to build a linkage model to connect data from different resolutions such that we could combine two-resolution data for final inferences.

At the same time, we need to handle the previously mentioned misalignment problem. The differences between two-resolution datasets are rigid body transformations plus some small distortions caused by resolution differences. The inter-point distances between high-resolution data points should be approximately the same as the distances between the corresponding low-resolution data points. So we utilize the

invariance of inter-point distances to match the two datasets, i.e., to find a "rough" correspondence between data from different resolutions. Then, based on the found pair-wise correspondence, we estimate the best rigid body transformations that bring two datasets together using an optimization.

The small distortions in both datasets, however, make it almost impossible to find the true match between the high-resolution and the low-resolution datasets. We address this issue from two aspects: First, we build a neighborhood linkage model which links each high-resolution data point with all the low-resolution data points in its neighborhood. This model can handle some small misalignments that remain after the matching procedure above. Second, we consider a number of possible matches and adopt Bayesian model averaging framework to assign a posterior probability to each match according to how well it supports observed data. Then the Bayesian model averaging produces combined final predictions of the surface.

Such a methodology, if successfully developed, will also have broad applications to other engineering domains, such as remote sensing. When surveying forest coverage, engineers could have both an scanning LIDAR (Light Detection and Ranging), which is of high-resolution but slow, and its fast counterparts, a low-resolution profiling LIDAR. Therefore, we would like to develop a general methodology that could potentially apply to multi-resolution problems in other engineering applications.

C. Outline of the Dissertation

Following the introduction, the dissertation is organized as follows. Chapter II provides a comprehensive literature review on existing methods related to combining data from different resolutions. It also explains how this dissertation relates to and differs from the existing methods.

Chapter III presents a Gaussian process model for single-resolution metrology data. The dissertation validates the single-resolution data model using both simulated data and real metrology data. This single-resolution model will later serve as a low-resolution data model in the context of combining two-resolution metrology data.

Chapter IV presents a Bayesian hierarchical model that combines misaligned metrology data from two different resolutions to better predict the part surface. It includes the following components: (a) a Gaussian process for modeling metrology data at the low-resolution level (the same single-resolution model presented in Chapter III); (b) a heuristic matching algorithm that produces a pool of candidate matches and their associated transformations between the two datasets; (c) a linkage model, conditioned on a given match, that connects a high-resolution data point to a set of low-resolution data points in its neighborhood and makes combined predictions; (d) finally, Bayesian model averaging of the predictive models in (c) over the pool of candidate matches found in (b). The Bayesian model averaging procedure assigns weights to different matches according to how well they support the observed data, and then produces the final combined prediction of the surface utilizing data from both resolutions. Chapter IV also presents the prior distributions for the model parameters.

Chapter V demonstrates the improvements of the proposed method over three alternative methods, using both simulated data and the datasets from a milled sine-wave part, measured by two coordinate measuring machines of different resolutions, respectively. The three alternative methods include observed low-resolution data, predictions using high-resolution data alone and predictions that combine two-resolution data but do not address the misalignment problem.

Chapter VI summarizes the dissertation and discusses some future research directions. The outline of this dissertation is shown in Fig. 3.

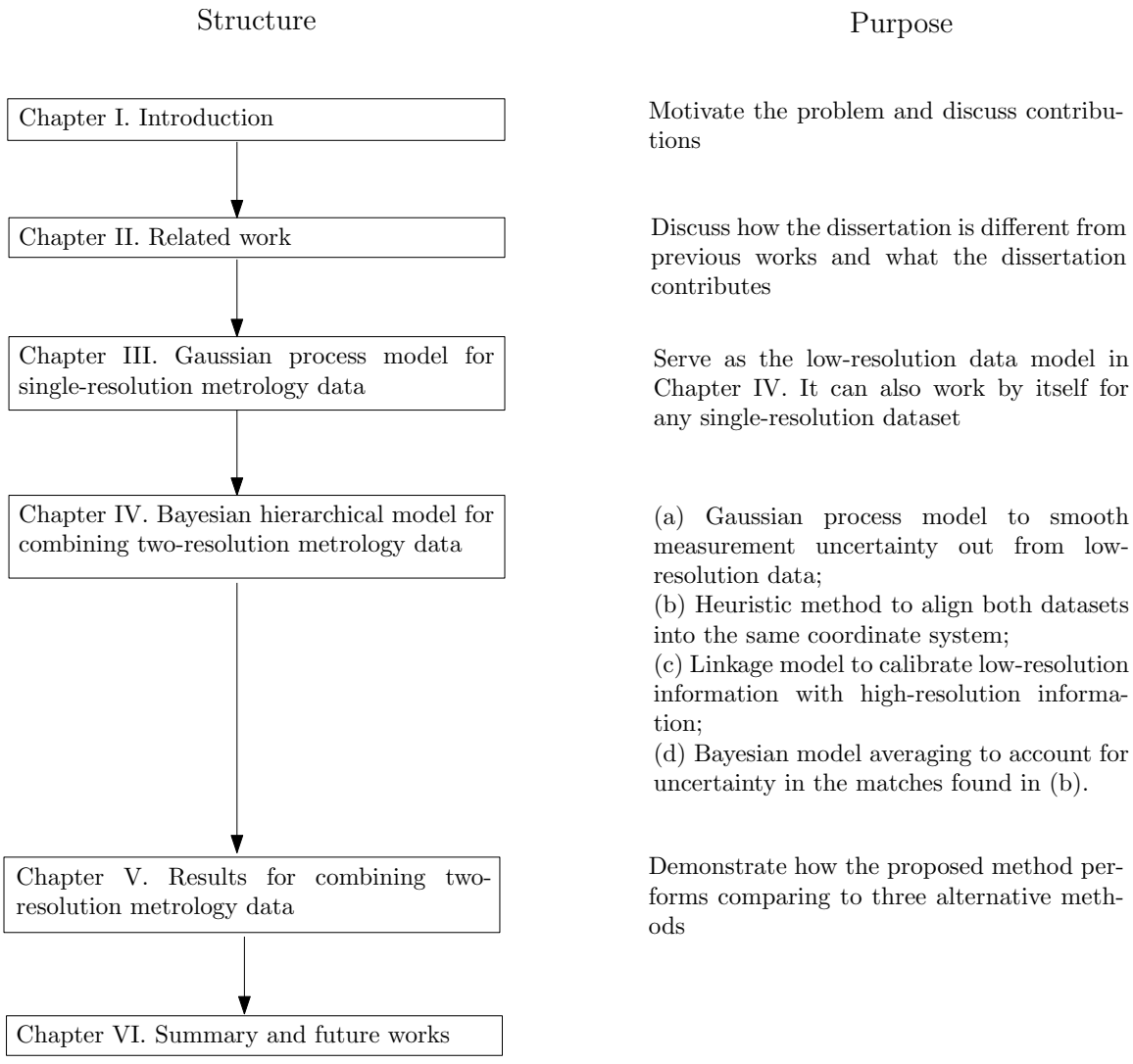

Fig. 3. Organization of the dissertation

CHAPTER II

RELATED WORK

This chapter reviews three categories of methodologies dealing with metrology data. The first category (Section II.A) deals with single-resolution metrology data. Even when data are from multiple metrology devices with different resolution levels, engineers sometimes simply put all the data together and apply the first category of methods, which basically does not differentiate data from different resolutions. The second and third categories of methods address the multi-resolution aspect of the data explicitly. The second category of methods uses data from different resolutions sequentially, but utilizes the high-resolution data alone for final inference instead of combining both datasets (Section II.B). The third category combines data from different resolutions using statistical models (Section II.C), but does not explicitly address the misalignment problem that we encounter in the metrology application (discussed in Chapter I Section A).

The dissertation also reviews the existing methods related to matching and aligning two datasets in Section II.D, which covers the methods from the literature of image registration and statistical shape analysis.

A. Methods Using Single-resolution Metrology Data

Reference [15] surveyed the related literature prior to 1997 that used single-resolution metrology data for the purpose of form error assessment. This survey paper discussed two major ideas: the minimum zone (MZ) method and the orthogonal least squares (OLS) method. The MZ method assumes that the limited metrology data represent the entire surface of a part well, finds the maximum inscribing and minimum circumscribing features that bound all the metrology data, and uses the calculated

orthogonal width to compare with a form tolerance. The OLS method assumes that a part's surface arises from a perfect geometric shape with independent and identically distributed (i.i.d.) errors. So it fits an ideal feature to metrology data by minimizing the sum of squared orthogonal residuals and uses the range of the resulting orthogonal residuals to compare with a form tolerance. Most papers surveyed in [15] provided different algorithms to implement the two ideas for various kinds of geometry features. Reference [15] also surveyed some variants of the OLS method, which use different objective functions, e.g., the least average deviation used in [50], [51] and [38].

After 1997, researchers continued searching for efficient and/or robust algorithms to implement the MZ or OLS method. More publications emerged on the MZ method than on the OLS method. Some proposed new ideas to find a MZ, such as the characteristic point-based method [12] or the minimum potential energy theory [18]. Others proposed exact MZ solutions of some geometric features [29], [6] (e.g., sphericity), which were not available before. Since it is computationally demanding to find the exact MZ solutions using traditional methods such as convex hull and Voronoi diagram, researchers reported on how to improve the efficiency to find the exact MZ [45], [46], [30] or on how to develop efficient approximation alternatives [53], [56], [64]. One method that caught much attention is the zone-fitting method [9], [10]. The zone-fitting method attempts to verify the form error conformance by transforming all the metrology data back to the design tolerance zone via an optimization routine. It is computationally less demanding than those expensive geometry methods and easier to implement for different types of features.

Reference [15] pointed out several open issues in estimating the surface of geometric features. One important issue is how to incorporate systematic manufacturing errors into a modeling and assessment procedure. Manufacturing errors are always stochastic in nature. In the literature, however, the random manufacturing errors

generally refer to the i.i.d. random deviations from an ideal form, while the systematic manufacturing errors refer to the non-i.i.d. deviations. Reference [15] gave a few examples of systematic manufacturing errors. It also pointed out that systematic manufacturing errors are common causes of form errors in reality. Therefore, a realistic model should definitely incorporate systematic errors. This issue has not been well addressed in the afore-referenced body of literature. Without accounting for systematic errors, the estimates of the part's surface and thereafter form error could significantly deviate from their actual values. In this dissertation, we develope a single-resolution model that attempts to capture the systematic errors using a Gaussian process model [1] (a spatial statistical model) in order to better estimate the form error.

Researchers have employed some methods to model the systematic errors. Some used a basis function method to approximate the systematic errors, e.g., a polynomial of different orders [63] or B-spline functions [61]. Some used Fourier analysis to distinguish the generally low-frequency systematic components from the high-frequency random components [27], [8], [13]. These models may require a large sample of metrology data to estimate the relatively many parameters involved in the polynomials or B-spline functions, or to enable a fine separation of frequencies in Fourier analysis.

Another line of methods used a spatial statistical model, typically, a Gaussian process model, to represent the systematic manufacturing errors [16], [62]. The non-parametric nature of Gaussian processes offers a better flexibility in learning general types of systematic errors than the basis-function fitting method and the Fourier analysis method. This nature makes a Gaussian process model more attractive, since people generally do not know before hand what the function form or the type of systematic manufacturing errors is in an actual geometric feature. However, research on the spatial model-based approach was much under-developed and not broadly

reported. References [16] and [62] used Gaussian process models. Their models require one coordinate variable to be an explicit function of the other two coordinate variables. Thus, their methods do not apply to generic geometric features, such as roundness, where one coordinate cannot be expressed as an explicit function of the other two.

This dissertation also develops a Gaussian process model for single-resolution metrology data, which continues this line of research. The proposed Gaussian process model works for generic geometric features, and improves form error assessment. Chapter III presents this Gaussian process model. The model does not require expressing one coordinate as an explicit function of other coordinates. It also incorporates both systematic errors (using a spatially correlated term) and random errors (using a spatially uncorrelated term). In contrast, References [16] and [62] did not consider spatially uncorrelated errors. This might not be reasonable provided that manufacturing processes produce both low-frequency systematic errors (because of machine tool wear) and high-frequency random errors (because of machine vibration). In addition, the Gaussian process model produces a distribution of the geometric surface, and uses it to estimate an empirical distribution of the form error. The empirical distribution reflects the estimation uncertainty (resulting from the sampling and modeling uncertainty) and can help quantify the risk of part acceptance. References [16] and [62] used one predicted surface to estimate the form error and decide on accepting or rejecting the part. They did not provide uncertainty information about the form error assessment.

B. Sequential Methods to Use Metrology Data from Different Resolutions

Using multiple-resolution metrology devices is a relatively new idea. Thus, research reported on combining the multiple datasets of difference resolutions is rather limited. There have been reports of CMMs carrying multiple types of sensors [7], [37], [49], [4], which typically include a mechanical touch probe and a vision system. A vision system has a much lower degree of resolution (between 100-200 microns [49]), which is close to, or sometimes even larger than, the magnitude of manufacturing errors. On the other hand, a vision system is highly efficient in capturing the global picture of the object under measurement. Thus, the goal of having both the mechanical probe and the vision system is as follows: use the vision system to locate the object and generate a rough product contour, and then to establish a sampling plan based on the product contour so that the touch probe can measure the product with little human guidance or intervention. The information from different sources is used sequentially. When it comes to predicting the product surface for quality assurance, only the high-resolution measurements (from the mechanical probe) are used. So the objective of this line of research is about improving the efficiency of a measurement procedure, while our research focuses on combining information from different sources for better predictions.

C. Statistical Methods to Combine Multi-resolution Data

Sophisticated methodologies have been developed to address the problem of synthesizing spatial data collected at different scales and resolutions [23], [58], [20], or the problem of calibrating (deterministic) computer simulation models of different accuracies or of calibrating computer simulations with physical measurements [33], [34], [43], [28], [40]. According to our survey, there are two major schools of thought.

The first idea is to establish respective models for data observed at individual resolution levels. The underlying true process serves as a linkage connecting data from different resolutions. Accordingly, the data from all the resolutions can be pooled together to infer about the true process. Most methods reviewed in [23], [58], and [43] adopted this approach. Of course, individual researchers had their own detailed treatments on top of the baseline. For example, Reference [43] treated the estimates of the parameters from the low-accuracy data as a prior to the high-accuracy data model and then updated the estimates using the high-accuracy data.

The second idea is to establish a single-resolution data model (typically, for low-resolution or low-accuracy data) and a linkage model connecting data from different resolutions. The linkage model assumes that each high-resolution response can be predicted by the *corresponding* low-resolution one after a scale change and a location shift. References [33], [34], [28], [20], and [40] adopted this approach. Again, different treatments were employed in the aforementioned literature: For example, Reference [20] used a Markov random field for the single-resolution data model, while all the others used a Gaussian process model.

We believe that the general ideas developed in the statistical literature are applicable to the metrology application. This dissertation follows the second idea as outlined above, that is, employing a low-resolution data model plus a linkage model. Chapter III develops a Gaussian process model for single-resolution metrology data of a manufactured product, which also serves as the low-resolution data model in the context of combining two-resolution data. Mathematically, the proposed single-resolution model is almost the same as the low-accuracy data model used in [40], except that it has an additional term representing the random measurement errors. More modeling details as well as physical interpretations are provided in Chapter III.

For the linkage model, we propose a *neighborhood linkage model* to link each

high-resolution output as a kernel regression of all the low-resolution information in its neighborhood. We let the size of the neighborhood be decided by data. Recall that Reference [40] mapped each high-accuracy data point to a single low-accuracy data point. Our neighborhood linkage model extends the one-to-one linkage to a one-to-many relationship. Because of this extension, the neighborhood linkage model is able to utilize all the high-resolution and low-resolution data, while the linkage model in [40] utilizes *all* the high-accuracy data plus a *small subset* of the low-accuracy data whose input settings are the same as those of the high-accuracy data.

The computer experiment literature, however, did not address the misalignment problem encountered in the metrology application. Although the computer simulation codes can be run at varying accuracy, the inputs are always designed precisely so that the correspondence relationship of inputs can be easily identified across different resolutions. The misalignment problem was not discussed in the problems of synthesizing spatial data, either, because the inputs for those problems are geographical locations, measurements of which, although contaminated by noises, are at least one magnitude more accurate than those of spatial responses. As a result, it is reasonable to assume the inputs in the spatial problems are also known precisely, the same as in the computer experiments.

D. Existing Methods on Matching and Aligning Datasets

The dissertation also explores the literature on matching and alignment methods (but please note that an alignment method does not address the prediction issue as required in our problem). The majority of the matching and alignment methods were developed for image analysis and registration applications. References [3] and [66] gave comprehensive reviews of the methods in image registration. Matching two ob-

jects or images refers to finding correspondences between points or features of one object and those of the other. This is also called labeling. After establishing the correspondences between two datasets, we can mark the matching relationship between them as labels; the data involved then become *labeled* points. Given the labeled points, the subsequent step is to estimate the transformation model to bring the two objects or images together, which is also called aligning. Statistical shape analysis [17] provides sophisticated methodologies that address the problem of estimating the transformation model and assessing the shape differences between different images based on labeled points.

Our problem of matching the metrology data points is to match *unlabeled* points. To find the pairwise matching correspondences of two sets of unlabeled data points, there are three major approaches in the image registration literature. One is to rely on the feature properties (e.g., curvature) associated with a point, or a graphic pattern formed by a set of points [66], [54]. In our metrology application, the high-resolution data points are very scarce due to the time and cost constraints. Therefore, we hardly get any feature property from high-resolution data, or we might obtain a different graphic pattern from high-resolution data than from low-resolution data. This makes this type of method does not work well to match scarce high-resolution metrology data with abundant low-resolution metrology data. The high-resolution points function more like a set of anchor points, and their contribution is to help calibrate the shape-revealing low-resolution points to the underlying true surface.

The other popular method of point matching is based on the invariance property of inter-point distances. The underlying principle is that using either dataset, the relative distance between the same two points should be approximately the same [42], [54]. The distance-based approach is applicable to the metrology application since data measured at different resolutions differ from one another by a rigid body trans-

formation plus small distortions caused by measurement errors. Thus, we will utilize the distance-based approach to match datasets from different resolutions.

Recently, Reference [25] developed a Bayesian method for finding matching and transformation between two configurations of points. The matching of points is represented by a matching matrix, whose (j, k)-th entry is 1 if the j-th point in one configuration matches the k-th point in another, and 0 otherwise. The transformation is represented by a rotation matrix and a translation vector. Reference [25] modeled the matching and aligning problem using a Bayesian approach, and their method estimates the matching and transformation results simultaneously.

Reference [25]'s method is highly relevant to our application because it handles unlabeled points, the same as what we need to do. We tried applying Reference [25]'s method to match and align the data points in the two-resolution datasets.

However, Reference [25]'s method performs poorly in matching and aligning the two-resolution metrology datasets. The reason is that we attempt to match sparse high-resolution data points (several tens) to dense low-resolution data points (several thousands). The parameter space of possible matches are high dimensional. An example used in Chapter V has high-resolution data of size 40 and low-resolution data of size 1,560 (i.e., $40 \times 15{,}60$). In contrast, the two examples in [25] have high-resolution data and low-resolution data of sizes 35×35 and 40×63. Exploring the high-dimensional parameter space with [25]'s procedure needs to run very long Markov Chain Monte Carlo (MCMC) chains, and is less likely to produce a good match given limited time. In addition, the matching problem in metrology also oftentimes involves matching 3D point clouds of symmetric shapes (e.g., cylinder and circles), which have been proved difficult due to the existence of numerous local optima [19]. So we cannot expect the Bayesian alignment model from [25] to converge to the global optima in a timely manner.

Thus, the dissertation chooses to develop a heuristic matching that utilizes the invariance property of inter-point distance to better suit our metrology application. The heuristic matching performs a greedy search and thus can explore the parameter space of matching much more efficiently. The heuristic matching algorithm finds a set of good matches, and the two-resolution model combines data from two resolutions averaging over different matches. The dissertation uses Bayesian model averaging to take care of the uncertainty in determining which match is the best one. The Bayesian model averaging automatically assigns weights to different matches according to how much they are consistent with the two-resolution model and the observed two-resolution data.

The existing literature on combining multi-resolution datasets is summarized in Table II.

Table II. Summary of existing methods on combining multi-resolution datasets

Section	Methodologies	Application domains	Research issues addressed	How is the dissertation different?
II.A	MZ method, OLS method and their variants [15]; Approximation with basis functions [63], [61]; Fourier analysis [27], [8], [13]; Spatial statistic model [16], [62]	Manufacturing quality control; Form error assessment	Provide a point estimate for form error using metrology data. Model systematic errors for some geometric features.	Provide a predictive distribution of form error, and therefore a probabilistic form error assessment. Address systematic errors and the method works for generic geometric features.
II.B	Sequential Methods [7], [37], [49], [4]	Multiple-sensor CMM; Information automation	Use low-resolution data to automate the measuring process of high-resolution sensor.	Combine information from two resolutions instead of discarding low-resolution data completely after obtaining high-resolution data.
II.C	Respective models for data from different resolutions [23], [58], [43]; Single-resolution model plus linkage [33], [34], [28], [20], [40]	Spatial data modeling; Computer experiment analysis	Combine multi-resolution data to final predictions when the correspondence between datasets from different resolutions is unknown.	Handle the misalignment problem explicitly
II.D	Statistical shape analysis [17]; Point matching for image registration [66], [54], [42]; Bayesian matching and alignment [25]	Image analysis; Bioinformatics	Estimate transform based on labeled points. Match unlabeled data points in a deterministic way. Account for uncertainties in matching unlabeled points and estimating transformation	Develop a heuristic matching to explore parameter space more efficiently. Adopt a Bayesian model averaging framework to account for uncertainty in matching. Combine two-resolution data for final predictions.

CHAPTER III

GAUSSIAN PROCESS MODEL FOR SINGLE-RESOLUTION DATA

A. General Idea of Single-resolution Model

When magnified, the surface of a manufactured part looks just like a geographical terrain. Please see Reference [52] for the topographies of a variety of machined surfaces. This analogy motivates applying a spatial statistical method to represent a manufactured geometric feature. Metrology measurements of a manufactured geometric feature can be decomposed into three portions (as shown in Fig. 4): a global trend portion, which follows the shape of the feature as defined by its designed form; a spatially correlated portion (when the systematic error exists, the measurements in close proximity on a surface show strong correlations); and a spatially uncorrelated portion, i.e., the random error portion. As such, we model the metrology measurements as arising from systematic errors and random errors added to an ideal geometric form.

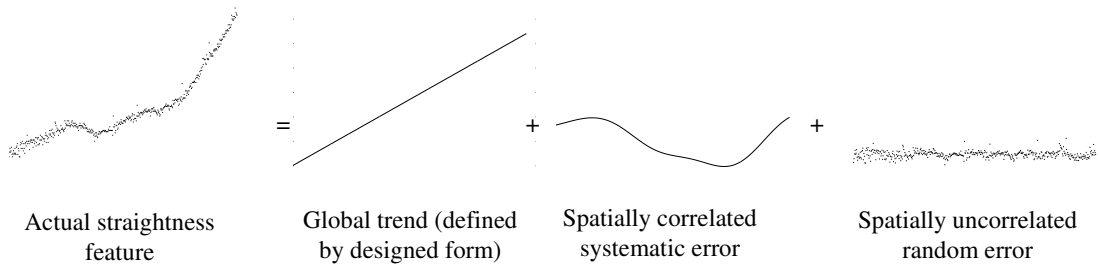

Fig. 4. Decomposition of metrology measurements of a manufactured straightness feature

B. Background: Gaussian Process Model

Gaussian process (GP) models are popular in modeling spatially correlated data. Some good references for Gaussian process models include References [48] and [47].

The key idea is to treat observed data as a realization from a Gaussian process (also called Gaussian random field). Suppose we are to study precipitation of a certain area S. Following a similar decomposition as in Fig. 4, we can decompose the spatially varying response into two components:

$$y(\mathbf{x}) = f(\mathbf{x}) + \xi(\mathbf{x}) \qquad \text{for } \forall\ \mathbf{x} \in S \tag{3.1}$$

where $f(\mathbf{x}_i)$ is the mean function representing the large-scale trend, and $\xi(\mathbf{x}_i)$ is a small-scale variation. The small-scale variation $\xi(\mathbf{x}_i)$ is assumed to be a Gaussian process with certain correlations. Through modeling the correlations between the responses from different locations, $R(\mathbf{x}_i, \mathbf{x}_j)$, we can capture the small-scale variation from data.

The commonly used correlation functions include two categories: isotropic correlation functions and anisotropic correlation functions. For isotropic correlation functions, the spatial correlation is a function solely of the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$. One frequently used isotropic correlation function is the Gaussian correlation model as follows:

$$R(\nu, \mathbf{x}_i - \mathbf{x}_j) = \exp\{-\nu||\mathbf{x}_i - \mathbf{x}_j||^2\} \tag{3.2}$$

where $||\cdot||^2$ denotes Euclidean distance and ν is the scale parameter controlling how quickly the correlation decays as the between-point distance increases.

For anisotropic correlation functions, the spatial correlation depends not only on the distance between locations, but also on the direction. A commonly used one is

the following:

$$R(\boldsymbol{\nu}, \mathbf{x}_i, \mathbf{x}_j) = \prod_{k=1}^{d} \exp\{-\nu_k (x_{ki} - x_{kj})^2\}, \tag{3.3}$$

where d is the dimension of the input variables, i.e., $\mathbf{x}_i = (x_{1i}, x_{2i}, \ldots, x_{di})$, and $\boldsymbol{\nu} = (\nu_1, \nu_2, \ldots, \nu_d)$ are the scale parameters controlling how fast the correlation decays as the between-input distance increases in each dimension.

As a result of the Gaussian process assumption, any number of the responses $\mathbf{y} = (y(\mathbf{x}_1), \ldots, y(\mathbf{x}_n))$ follow a multivariate normal distribution

$$\mathbf{y} \sim N(\mathbf{f}, \boldsymbol{\Sigma}), \tag{3.4}$$

where $\boldsymbol{\Sigma} = \kappa^2 \mathbf{R}$, $\mathbf{R}$ is the correlation matrix with its (i, j)-th entity defined as in either (3.2) or (3.3), and κ^2 is the variance for the small-scale trend $\xi(\cdot)$.

Here we illustrate how a GP model works. To simplify the illustration, we consider a one-dimensional area. Suppose we observe some precipitation data illustrated by crosses in Fig 5 (a). The grey line is the underlying true process.

We assume that the observed data follow Model (3.1) with the mean function $f(\mathbf{x})$ equaling to zero, and estimate the scale parameter and the variance (Fig. 5 (b)). Based on the estimated value, we can predict the entire process as in Fig. 5 (c), where the predictive mean is the black line, and the 95% predictive band is the grey region around the black line.

For spatial data, the inputs $\mathbf{x}$ are usually geographic locations and the responses y are observations of some spatial phenomena, for example, precipitation or temperature. In the metrology problem, the observed data are three-dimensional coordinate measurements of a part's surface. References [16] and [62] applied GP models to the metrology application before this dissertation. Their GP models used one or two coordinate variable, e.g., u and/or v, as its inputs, and the remaining coordinate,

e.g., w, as its response. So their GP models require one coordinate variable to be an explicit function of the other coordinate variables. This implies that their models can not apply to a general geometric feature, e.g., a circular feature, where the v coordinate is not an explicit function of the other coordinate u. In order to make the GP models to be applicable to generic geometric features, we have to redefine the input and the response variables.

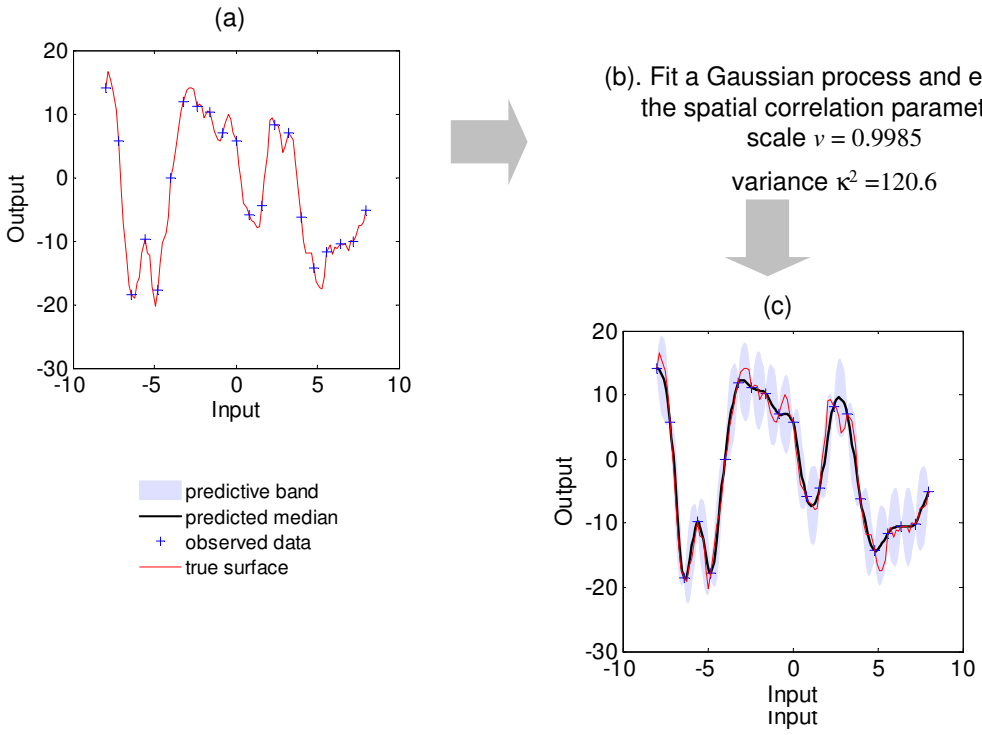

Fig. 5. Illustration of how Gaussian process model works

C. Gaussian Process Model for Single-resolution Metrology Data

First, we explain the measuring mechanism of a computer-controlled CMM, and how it inspires which variable to use as the input and which variable to use as the out-

put/response for the GP model. The machine takes the nominal position to be measured, $\mathbf{x}_i$, from an operator (via a computer interface) or a database storing the predetermined position information. The nominal position $\mathbf{x}_i$ is usually a point on an "imaginary" nominal geometry, as illustrated in Fig. 6. The the machine directs its measuring probe or laser beam to travel in a certain direction, denoted by $\mathbf{p}_i$, toward the object to be measured. Finally it retrieves the coordinate information, denoted by $\mathbf{a}_i$, of a point on the actual manufactured feature. This response that a machine returns, $\mathbf{a}_i$, is the point on the actual surface, by which a laser beam is reflected or a probe is stopped, shown as in Fig 6. The directional vector $\mathbf{p}_i$, associated with each measurement taken, is either specified by the operator or calculated by the computer automatically. This mechanism indicates that the input is $\mathbf{x}_i$ and the response is $\mathbf{a}_i$. Essentially, we map all the points on the nominal geometry to the actual manufactured feature. The GP model is set up to capture the mapping between these two surfaces. The advantage is that when we use the nominal position $\mathbf{x}_i$ as the input variable, it works for any type of geometric features and thus avoids the restriction in the previous GP models.

The response $\mathbf{a}_i$ is typically a vector: for example, for a three dimensional feature, $\mathbf{a}_i = (a_{u_i}, a_{v_i}, a_{w_i})^T$. We avoid using (x, y, z) as the coordinate variables because x and y have been used in the model for different meanings. A multivariate GP model is obviously more difficult to handle than a univariate model. Thus we want to further reduce the multivariate response into a one-dimensional response variable. Here we adopt the strategy used by [32], which suggested projecting the value of $\mathbf{a}_i$ onto the probe approaching direction $\mathbf{p}_i$ (usually the norm direction to the local surface) and using the resulting scalar as the response. As such, the GP model for single-resolution

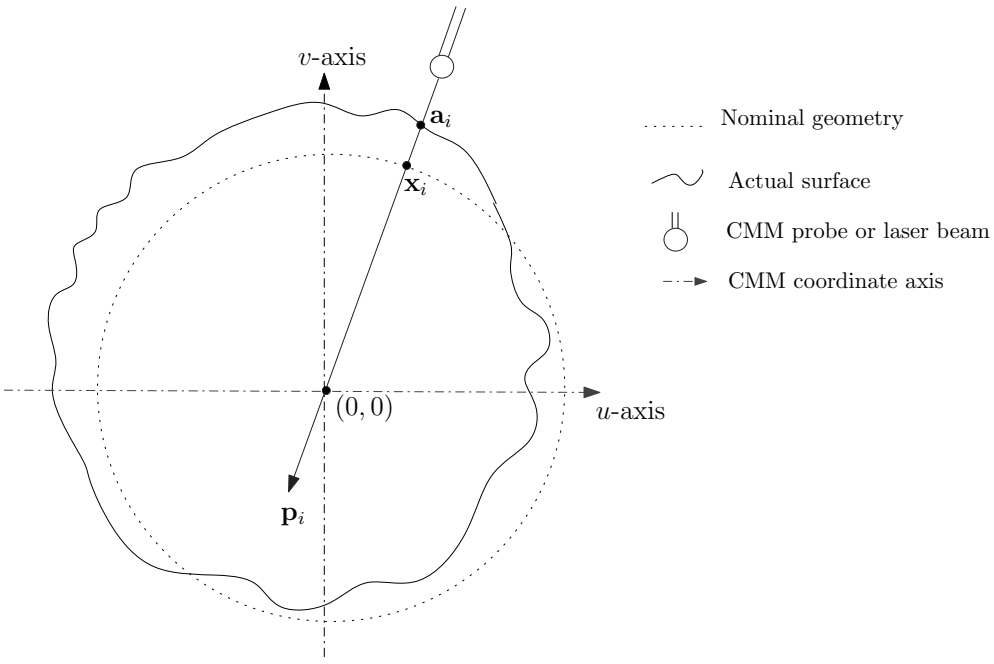

Fig. 6. Input and Output of the coordinate measuring process

metrology data is:

$$y_i \equiv \mathbf{a}_i^T \mathbf{p}_i = \mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})^T \mathbf{p}_i + \xi(\mathbf{x}_i) + \epsilon, \tag{3.5}$$

where y_i is the i-th metrology measurement projected onto the $\mathbf{p}_i$ direction, $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ corresponds to a geometry which has the ideal geometric form that engineers design, $\xi(\mathbf{x}_i)$ is the systematic error modeled by a spatially correlated term, and ϵ is the random error modeled by the spatially uncorrelated term. Generally, the random error includes both random manufacturing errors and measurement noises. When the metrology data are from the high-resolution level (i.e., high accuracy and low noises), we could assume that the measurement errors are negligible and then attribute the second error term ϵ entirely to the random manufacturing errors. When the metrology data are from the low-resolution level (i.e., low accuracy and high noises), the second

error term ϵ is mainly measurement noises.

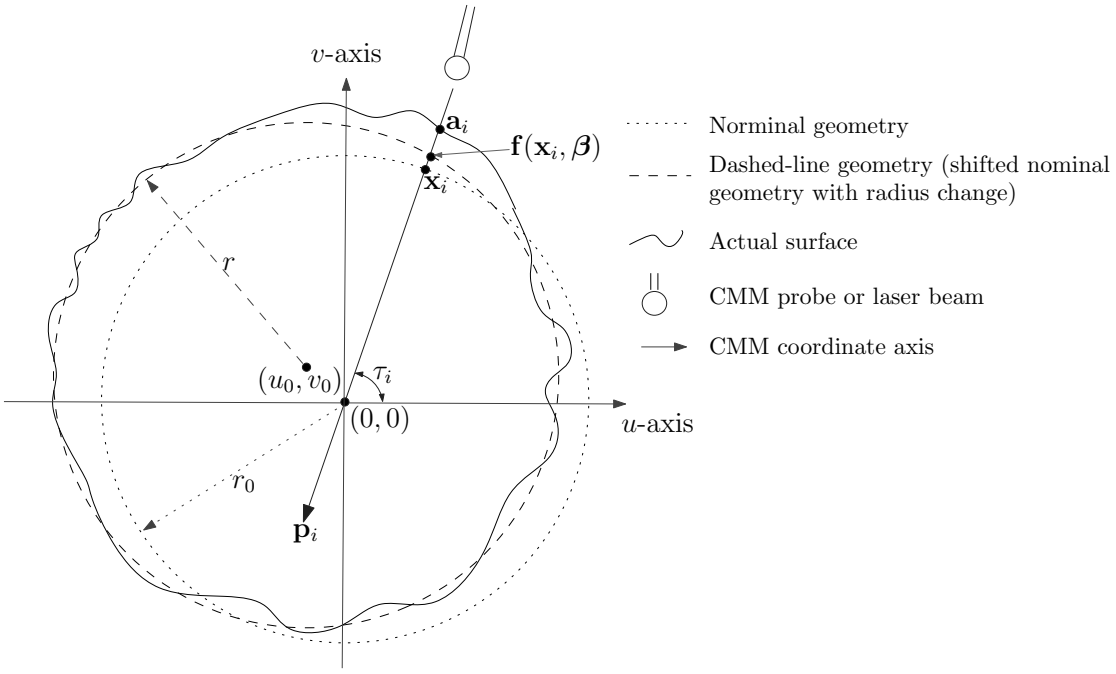

Fig. 7. Illustrations for GP model

The vector $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ represents the point on a dashed-line geometry that corresponds to $\mathbf{x}_i$ on the nominal geometry, shown as in Fig 7. The dashed-line geometry is a shifted nominal geometry that also includes possible small dimensional changes. So $\mathbf{f}(\cdot, \cdot)$ has the same function form as the nominal geometry does and is known from the part's design. The value of $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ incorporates the actual position of the part, which may undergo a rigid body motion during the fixturing process, as well as some small changes in dimensions of the part caused by manufacturing errors. In other words, $\boldsymbol{\beta}$ includes two factors: $\boldsymbol{\beta} = (\boldsymbol{\varphi}, \boldsymbol{\psi})$, where $\boldsymbol{\varphi}$ includes the parameters characterizing the rigid-body motion (e.g., the shift of the center of a round part), and $\boldsymbol{\psi}$ includes the dimension parameters (e.g., the radius of a round part). Therefore, $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ can incorporate dimensional errors, for instance, $\boldsymbol{\psi}$ can denote the radius

value of an actual part, which may be different from the design value.

Fig. 7 exaggerates the differences between the dashed-line geometry and the nominal geometry for the purpose of illustration. Typically, the actual solid-line surface follows very closely to the dashed-line geometry, and the differences are no more than a few hundreds of microns. The closeness between the dashed-line geometry and the actual surface motivates selecting $\mathbf{f}(\cdot, \boldsymbol{\beta}_l)$ as the mean component for the GP model. A three-step procedure to compute the $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ for a general geometry is provided in Section III.D.

In Model (3.5), the random error ϵ is modeled as i.i.d. $N(0, \sigma_\epsilon^2)$. The systematic error $\xi(\cdot)$ is assumed to be a Gaussian process independent of ϵ , and of zero-mean and covariance function $cov(\xi(\mathbf{x}_i), \xi(\mathbf{x}_j)) = \kappa^2 \cdot R(\boldsymbol{\nu}, \mathbf{x}_i - \mathbf{x}_j)$, where $R(\boldsymbol{\nu}, \cdot)$ is the correlation function with hyper-parameter $\boldsymbol{\nu}$. The rationale behind is that the systematic departures from the ideal geometric shape can be regarded as a sample path of a suitably chosen Gaussian process $\xi(\cdot)$.

The choice of the correlation function $R(\boldsymbol{\nu}, \cdot)$ should reflect the characteristics of the systematic manufacturing errors. It should also account for the amount of the data. When the amount of data is low, it would not serve us well if the GP model involves too many parameters to be estimated. Under this situation, we could adopt the Gaussian correlation function as in (3.2), which is isotropic and widely used in spatial statistics. The Gaussian correlation function has only one unknown parameter. Past experiences indicate that this correlation function has a good capability in modeling various spatial features [65], [21]. The experiences also indicate that the Gaussian correlation function appears reasonable for a number of manufactured geometric features. When the data suggest that the spatial correlation are significantly different from (3.2) for some particular part/feature, the isotropic assumption can be relaxed by including different parameters to control the spatial correlation scale for

different directions, for example, (3.3). The correlation function (3.3) involves more parameters, and therefore it requires more measurements to estimate them well.

There have been some research reports on modeling more general variance-covariance structures of spatially correlated measurements. For example, Reference [5] proposed a correlation model for lattice-structured metrology measurements on a part's surface. It assumed that the overall correlation is the product of the row-wise and the column-wise correlations, and that the metrology measurements in a row or column follow an autoregressive moving average (ARMA) process. This line of research is valuable for modeling the systematic manufacturing errors as a proper variance-covariance structure is essential to a Gaussian process model. Under the circumstance that the aforementioned correlation functions do not fit data well, the correlation structure proposed in [5] could be a good alternative.

The parameters for the GP model are $\boldsymbol{\theta} = (\boldsymbol{\beta}, \kappa^2, \sigma_\epsilon^2, \boldsymbol{\nu})$. In practice, engineers use a CMM to measure m data points $\{\mathbf{x}_i, y_i\}_{i=1,\ldots,m}$ from one part, and use them to estimate the unknown parameters $\boldsymbol{\theta}$. Plugging the estimated values for the parameters $\boldsymbol{\theta}$ into the GP model, engineers can use the model to predict the actual coordinate at any unmeasured location and to reconstruct the entire geometric feature. Finally, engineers can assess the form error of the geometric feature using the reconstructed geometric feature. The overall procedure is shown in Fig. 8, where $\mathrm{GP}(\cdot, \cdot)$ denotes a Gaussian process with the specified mean and the covariance matrix.

D. Determine the Ideal Geometric Form

In order to fully specify the GP model in Equation (3.5), this section presents a procedure to determine the ideal geometric form $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})^T \mathbf{p}_i$ for generic geometric features. The procedure follows an idea first proposed by Reference [32] when discussing a Man-

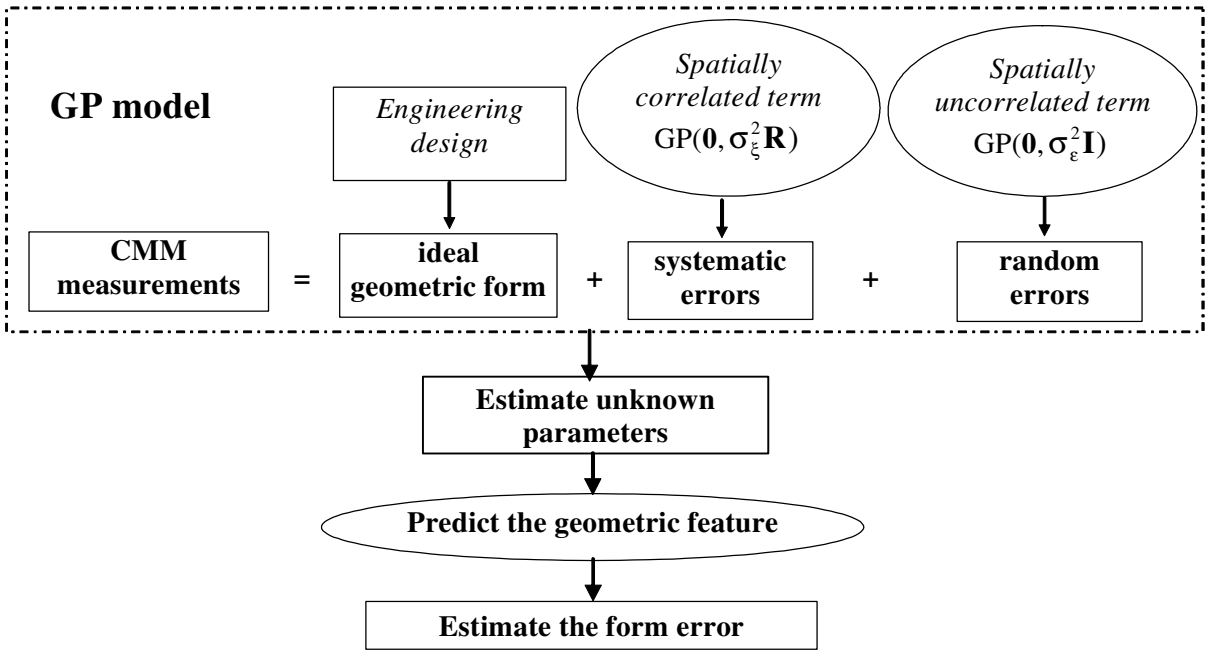

Fig. 8. Procedure of the GP modeling for single-resolution measurements

ufacturing Part Model (MPM). We repeat some of [32]'s original description here in order to make this dissertation a self-contained piece and to provide a basis for later discussions. Consider the process that a CMM measures a manufactured part with a general geometry shape, represented by $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ where $\boldsymbol{\varphi}$ and $\boldsymbol{\psi}$ follow the same meaning as we explained in Section III.C. Denote the nominal design values for the dimension parameters by $\boldsymbol{\psi}^*$. As such, $M(\mathbf{0}, \boldsymbol{\psi}^*)$ represents the designed feature that is of nominal sizes and is perfectly aligned with the CMM reference coordinate. Normally, $M(\mathbf{0}, \boldsymbol{\psi}^*)$ serves as the nominal geometry. $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ has the same geometry as $M(\mathbf{0}, \boldsymbol{\psi}^*)$ but differs from it in terms of a rigid body motion and some dimension change. Therefore, $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ represents the ideal form, the same as $\mathbf{f}(\cdot, \cdot)$. More specifically, $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ is a point on $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ (recall that $\boldsymbol{\beta} = (\boldsymbol{\varphi}, \boldsymbol{\psi})$). As mentioned in Section III.C, for a given part, the function form of $M(\cdot, \cdot)$ or that of $\mathbf{f}(\cdot, \cdot)$ is known

from the computer-aided design model.

During a coordinate measuring process, $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ slightly deviates from $M(\mathbf{0}, \boldsymbol{\psi}^*)$ in both location and dimension sizes. The dimension aspect is easy to understand since a manufacturing process supposedly produces the required dimensions with reasonable accuracy.

The location aspect can be understood as follows. A CMM can set up its reference coordinate through *soft fixturing* [31]. Before a metrology device performs the actual measuring task, it will first undergo a soft-fixturing process. That is, the machine takes a few measurements from the actual surface and uses them to estimate where the part is located. Since the actual dimension and shape of the part is unknown at the moment, the measurement machine must decide the location of the part assuming that the part has the design function form, e.g., a perfectly round circle for Fig. 9, and dimensions, e.g., the design value for radius r_0 for the round part in Fig. 9. Then the machine translates its origin $(0, 0)$ to the center of the part. Now the machine has a known geometry, e.g., the circle centered at $(0, 0)$ with radius r_0 in Fig. 9, serving as the reference for the subsequent measuring process. It is denoted by the dotted line $M(\mathbf{0}, \boldsymbol{\psi}^*)$ and is often called the *nominal geometry*. When the metrology device takes an input position $\mathbf{x}_i$, it considers this point to be on the nominal geometry. Because soft-fixturing only takes a few measurements to estimate the object's location, the estimation is not perfect, but the estimation errors are small. As a result, $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ also deviates from $M(\mathbf{0}, \boldsymbol{\psi}^*)$ in both location and dimension only slightly.

The differences between the actual surface (solid-line) and the dashed-line geometry $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ are the *form or geometric* errors. They are also caused by manufacturing errors. Fig. 10 summarizes all the three types of errors that we explain. Fig. 9 exaggerates the differences for the purpose of illustration. Typically, the actual surface follows the dashed-line geometry closely, and the differences are no more than a few

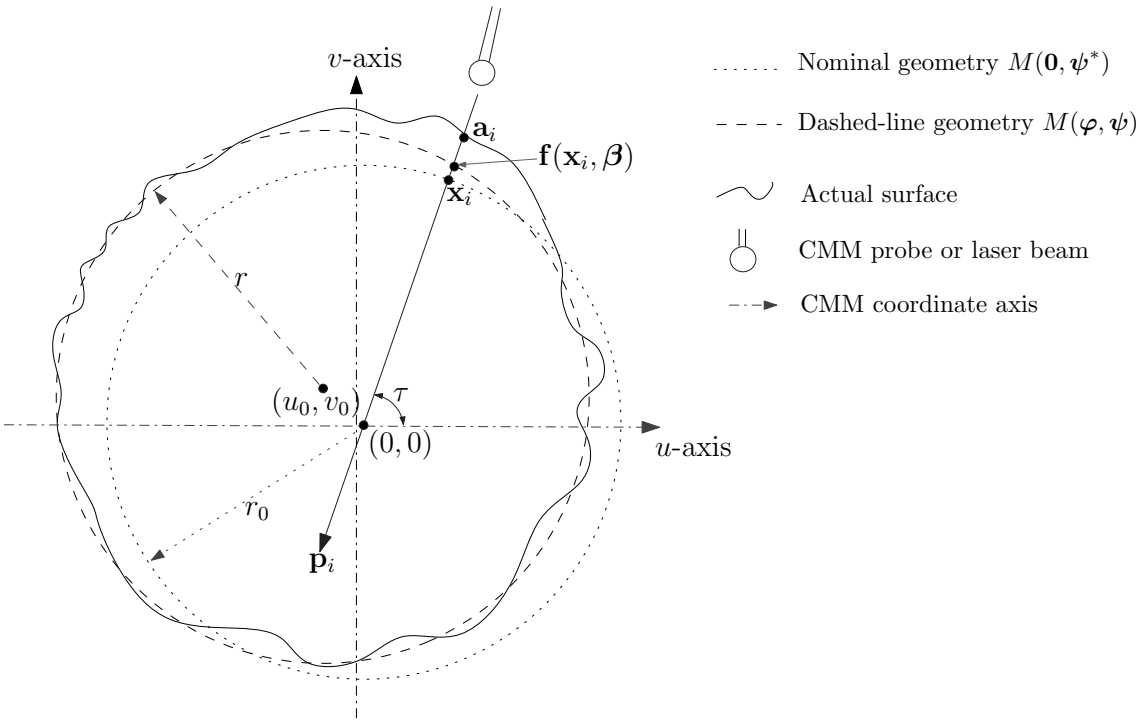

Fig. 9. Illustration of the coordinate measuring process

hundreds of microns. The closeness between the dashed-line geometry and the actual surface motivates selecting $\mathbf{f}(\cdot, \boldsymbol{\beta}_l)$ as the mean component for the GP model.

Determining $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ requires solving some geometry equations for the given point $\mathbf{x}_i$. When a CMM is directed to measure a point $\mathbf{x}_i$, it will automatically calculate its path, an approaching direction $\mathbf{p}_i$, based on its knowledge of the nominal surface. The approaching vector $\mathbf{p}_i$ is usually the norm direction to the nominal surface $M(\mathbf{0}, \boldsymbol{\psi}^*)$ as the true shape and location is never known. When it moves along, the probe will touch the actual manufactured surface and return the measurement value $\mathbf{a}_i$. Geometrically, $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ is an intersection point of the geometry shape $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ and the line passing through $\mathbf{x}_i$ and $\mathbf{x}_i + \mathbf{p}_i$. Therefore, both $\mathbf{a}_i$ and $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ lie on the line going through $\mathbf{x}_i$ and $\mathbf{x}_i + \mathbf{p}_i$. If projecting both $\mathbf{a}_i$ and $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ along $\mathbf{p}_i$, one

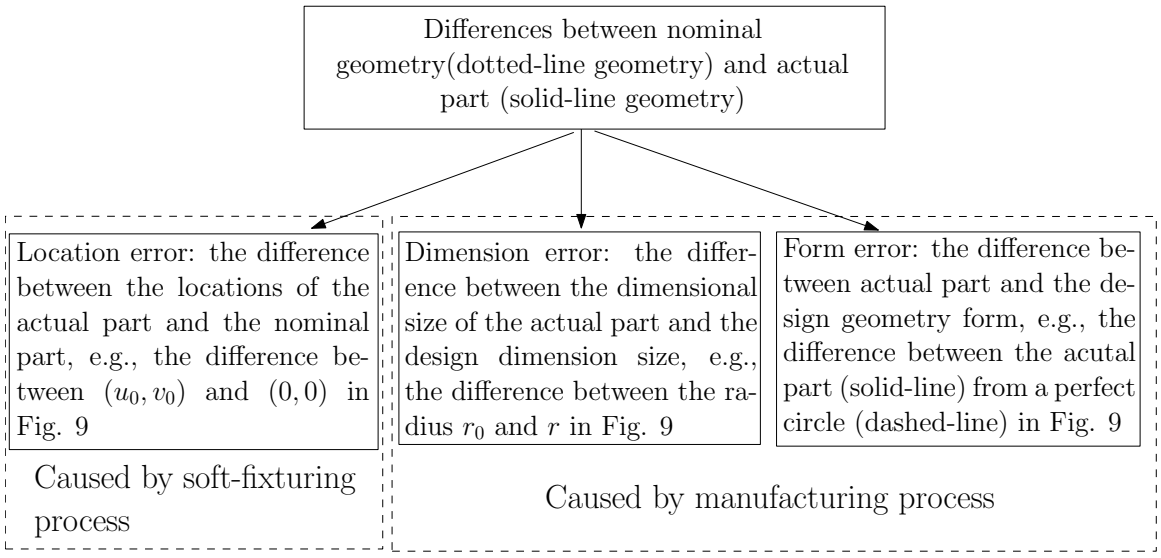

Fig. 10. Three different types of errors

will end up with the univariate GP model in (3.5).

The above discussion outlines how to decide $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})^T \mathbf{p}_i$. No general analytical formula, however, can be devised for an arbitrary geometry. Engineers will have to go through the following procedure (please refer to Fig. 11 for illustrations):

(i). Decide the approaching direction $\mathbf{p}_i$ according to $M(\mathbf{0}, \boldsymbol{\psi}^*)$ and $\mathbf{x}_i$, and decide the line function passing through $\mathbf{x}_i$ and $\mathbf{p}_i$.

(ii). Solve for the intersection point(s) between the line function and $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$.

(iii). If there are more than one intersection points, select the first intersection point when traveling with the CMM probe toward $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$.

(iv). Then, $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})^T \mathbf{p}_i$ is simply the vector inner product of the coordinates of the intersection point $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ and $\mathbf{p}_i$.

In the following subsections, we will illustrate how to implement the procedures to determine the ideal geometric form for two commonly used geometric features, straightness and roundness.

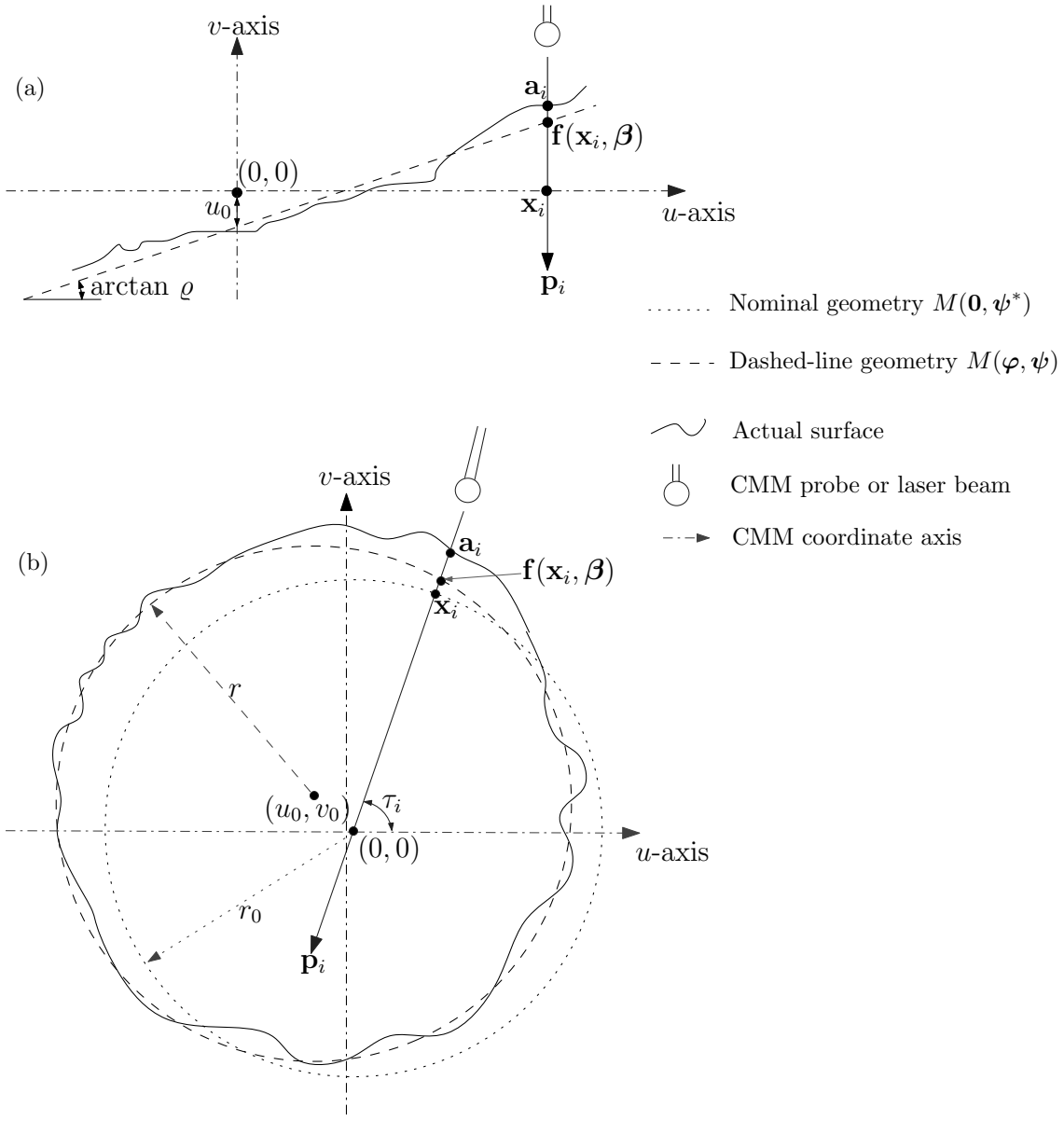

Fig. 11. Demonstrations of deciding the ideal geometric form

1. Straightness feature

For a straightness feature, as illustrated in Fig. 11 (a), the nominal geometry $M(\mathbf{0}, \boldsymbol{\psi}^*)$ is $v = 0$. The general geometry shape $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ is $v = v_0 + \varrho u$, where $\boldsymbol{\varphi} = (v_0, \varrho)$ is the location parameter with v_0 representing translation and ϱ representing rotation. For a straight line, the dimension parameter $\boldsymbol{\psi}$ degenerates. So $\boldsymbol{\beta} = (v_0, \varrho)$. For a point $\mathbf{x}_i = (u_i, 0)^T$ on the nominal feature $v = 0$, the approaching direction is $\mathbf{p}_i = (0, -1)^T$. After solving for the intersection between $v = v_0 + \varrho u$ and the vertical line $u = u_i$ and then calculating the vector inner product in Step (iv), we have $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})^T \mathbf{p}_i = (u_i, v_0 + \varrho u_i) \cdot (0, -1)^T = -v_0 - \varrho u_i$.

2. Roundness feature

For a straightness feature, v is an explicit function of u. But this is not the case for a roundness feature. Denote the roundness feature's nominal radius by r_0 and its actual radius by r. As illustrated in Fig. 11 (b), the nominal geometry $M(\mathbf{0}, \boldsymbol{\psi}^*)$ is $u^2 + v^2 = r_0^2$ and the general geometry shape $M(\boldsymbol{\varphi}, \boldsymbol{\psi})$ is $(u - u_0)^2 + (v - v_0)^2 = r^2$. The location parameter $\boldsymbol{\varphi}$ only consists of the translation of the center, i.e., (u_0, v_0), because a roundness feature is invariant under rotation. The dimension parameter $\boldsymbol{\psi}$ is the radius r. So $\boldsymbol{\beta} = (u_0, v_0, r)$. Given the nominal feature $M(\mathbf{0}, \boldsymbol{\psi}^*)$, we will have $\mathbf{x}_i = (r_0 \cos \tau_i, r_0 \sin \tau_i)^T$ and $\mathbf{p}_i = (-\cos \tau_i, -\sin \tau_i)$, where τ_i is the polar angle. Consequently, $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})$ is an intersection point of $(u - u_0)^2 + (v - v_0)^2 = r^2$ and the line passing through $(0, 0)$ and $\mathbf{x}_i = (r_0 \cos \tau_i, r_0 \sin \tau_i)$, i.e., a solution of the following equations:

$$\begin{cases} (f_{u_i} - u_0) + (f_{v_i} - v_0)^2 = r^2 \\ \frac{f_{v_i}}{f_{u_i}} = \tan \tau_i \end{cases} \tag{3.6}$$

where (f_{u_i}, f_{v_i}) are the two coordinates of the intersection points. Notice that a circle has two intersection points with a line. Pick the first intersection point according to the approaching path of the probe as the final solution (f_{u_i}, f_{v_i}). In this way, we can numerically calculate $\mathbf{f}(\mathbf{x_i}, \boldsymbol{\beta})^T \mathbf{p}_i = -f_{u_i} \cos \tau_i - f_{v_i} \sin \tau_i$ for any given $\mathbf{x}_i$.

For the value of $\mathbf{f}(\mathbf{x_i}, \boldsymbol{\beta})^T \mathbf{p}_i$, we can also calculate it via a geometrical method for roundness features: As illustrated in Fig. 12, $\mathbf{f}(\mathbf{x_i}, \boldsymbol{\beta})^T \mathbf{p}_i$ is the t_i in a triangle constructed from the three points: $\mathbf{f}(\mathbf{x_i}, \boldsymbol{\beta})$, (u_0, v_0) and $(0, 0)$. A similar triangle was utilized in [55]. Solving this triangle, we get $\mathbf{f}(\mathbf{x_i}, \boldsymbol{\beta})^T \mathbf{p}_i = -u_0 \cos \tau_i - y_0 \sin \tau_i - \sqrt{r^2 - (u_0 \sin \tau_i - v_0 \cos \tau_i)^2}$.

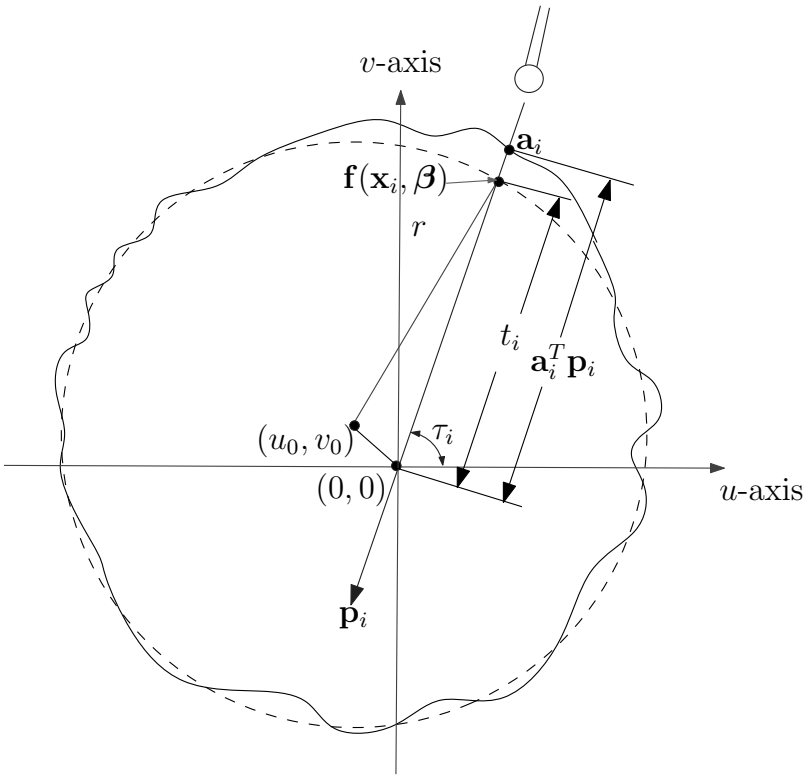

Fig. 12. Triangle relationship in a roundness feature

E. Predictive Distribution and Probabilistic Form Error Assessment Procedure

1. Maximum likelihood estimate for parameter estimation

Recall that the parameters in the GP model (3.5) are $\boldsymbol{\theta} = (\boldsymbol{\beta}, \kappa^2, \sigma_\epsilon^2, \boldsymbol{\nu})$, and the observed data are $\{\mathbf{x}_i, y_i\}_{i=1,\ldots,m}$. We arrange all the y_i in $\mathbf{y}_o$, i.e., $\mathbf{y}_o = (y_i)_{i=1,\ldots,m}$ where the subscript "o" implies the observed metrology measurements. We arrange all the $\mathbf{x}_i$ likewise, i.e., $\mathbf{X}_o$ is defined as a $m \times d$ matrix whose i-th row is $\mathbf{x}_i$. Given the Gaussian process assumption, the distribution of $\mathbf{y}_o$ given $\boldsymbol{\theta} = (\boldsymbol{\beta}, \kappa^2, \sigma_\epsilon^2, \boldsymbol{\nu})$ is:

$$(\mathbf{y}_o|\boldsymbol{\theta}) \sim N(\mathbf{g}(\mathbf{X}_o, \boldsymbol{\beta}), \boldsymbol{\Sigma}_o), \tag{3.7}$$

where $\mathbf{g}(\mathbf{X}_o, \boldsymbol{\beta})$ is an $m \times 1$ vector whose i-th element is $g(\mathbf{x}_i, \boldsymbol{\beta}) = \mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta})^T \mathbf{p}_i$, and $\boldsymbol{\Sigma}_o = \kappa^2 \mathbf{R}_o + \sigma_\epsilon^2 \mathbf{I}$ and $\mathbf{R}_o$ is an $m \times m$ matrix whose (i, j)-th elements is defined as either in Equation (3.2) or (3.3).

We employ a maximum likelihood estimator (MLE) to estimate the parameters in the GP model. Based on Distribution (3.7), the log-likelihood function for $\mathbf{y}_o$ will be:

$$\begin{aligned} 2l(\boldsymbol{\beta}, \kappa^2, \sigma_\epsilon^2, \boldsymbol{\nu}) = &- \log(\det(\kappa^2 \mathbf{R}_o + \sigma_\epsilon^2 \mathbf{I})) \\ &- (\mathbf{y}_o - \mathbf{g}(\mathbf{X}_o, \boldsymbol{\beta}))^T (\kappa^2 \mathbf{R}_o + \sigma_\epsilon^2 \mathbf{I})^{-1} (\mathbf{y}_o - \mathbf{g}(\mathbf{X}_o, \boldsymbol{\beta})) - m \log(2\pi) \end{aligned} \tag{3.8}$$

An MLE of $\boldsymbol{\theta} = (\boldsymbol{\beta}, \kappa^2, \sigma_\epsilon^2, \boldsymbol{\nu})$ can be obtained by maximizing $l(\boldsymbol{\beta}, \kappa^2, \sigma_\epsilon^2, \boldsymbol{\nu})$, i.e., $\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} l(\boldsymbol{\beta}, \kappa^2, \sigma_\epsilon^2, \boldsymbol{\nu})$. We solve it using a gradient-based optimization routine in `MATLAB`. Other similar optimizers can fulfill the task as well. Two more actions are taken to improve the optimization efficiency. One is that we substitute three parameters $(\kappa^2, \sigma_\epsilon^2, \boldsymbol{\nu})$ with $(\exp(\omega_1), \exp(\omega_2), \exp(\boldsymbol{\omega}_3))$. This is because the variance parameters κ^2 and σ_ϵ^2, and the correlation hyper-parameter $\boldsymbol{\nu}$ can only take posi-

tive values, and it makes a constrained optimization. After the transformation, we deal with an unconstrained optimization, which is generally easier to solve. For an unconstrained optimization, we use the `MATLAB` function *fminunc*.

For a straightness feature, its GP model has a linear mean structure. Thus, we can use a Restricted Maximum Likelihood (REML) estimator, which is supposed to give less biased estimates of the covariance parameters (i.e., κ^2, σ_ϵ^2, and $\boldsymbol{\nu}$) than MLE. Reference [11] proved the asymptotic property of REML estimation for the GP covariance parameters. Reference [59] gave the details for implementing a RMLE for a linear GP model. For a roundness feature, Model (3.5) has a nonlinear mean structure, and REML estimation is not available. Therefore, we use an MLE for roundness features.

2. Predictive distribution of geometric surface and form error assessment

For the purpose of form error assessment, we are ultimately interested in predicting the behavior of the geometric feature based on the observed metrology data. We approximate the continuous surface of a geometric feature by a dense set of points on it. Denote the dense set of nominal points on a geometric feature by $\{\mathbf{n}_i\}_{i=1:N}$, and it is deemed representative for the feature. The prediction of the geometry at $\mathbf{n}_i$ is $y(\mathbf{n}_i)$. We arrange all N predictions in a vector $\mathbf{y}_\mathrm{p}$, i.e., $\mathbf{y}_\mathrm{p} = (y(\mathbf{n}_i))_{i=1:N}$, where the subscript "p" implies a prediction. Given the Gaussian process assumption, $\mathbf{y}_\mathrm{p}$ and $\mathbf{y}_\mathrm{o}$ follow a joint multivariate normal distribution:

$$(\mathbf{y}_\mathrm{p}, \mathbf{y}_\mathrm{o}|\boldsymbol{\theta}) \sim N\left(\begin{bmatrix} \mathbf{g}(\mathbf{X}_\mathrm{p}, \boldsymbol{\beta}_l) \\ \mathbf{g}(\mathbf{X}_\mathrm{o}, \boldsymbol{\beta}_l) \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_\mathrm{p} & \boldsymbol{\Sigma}_\mathrm{po} \\ \boldsymbol{\Sigma}_\mathrm{po}^T & \boldsymbol{\Sigma}_\mathrm{o} \end{bmatrix}\right), \tag{3.9}$$

where $\mathbf{g}(\mathbf{X}_\mathrm{o}, \boldsymbol{\beta})$ and $\boldsymbol{\Sigma}_\mathrm{o}$ are defined in Equation (3.7); $\mathbf{g}(\mathbf{X}_\mathrm{p}, \boldsymbol{\beta})$ is an $N \times 1$ vector whose i-th element is $g(\mathbf{n}_i, \boldsymbol{\beta}) = \mathbf{f}(\mathbf{n}_i, \boldsymbol{\beta})^T \mathbf{p}_i$; $\boldsymbol{\Sigma}_\mathrm{po}$ is the covariance matrix between

$\mathbf{y}_\mathrm{p}$ and $\mathbf{y}_\mathrm{o}$, $\mathbf{\Sigma}_\mathrm{po} = \kappa^2\mathbf{R}_\mathrm{po} + \sigma_\epsilon^2\mathbf{E}$, and the matrix $\mathbf{E}$ has its (i,j)-th elements $E_{ij} = 1$ if $\mathbf{n}_i = \mathbf{x}_j$ and 0 otherwise; $\mathbf{\Sigma}_\mathrm{p}$ is the covariance matrix of $\mathbf{y}_\mathrm{p}$ and $\mathbf{\Sigma}_\mathrm{p} = \kappa^2\mathbf{R}_\mathrm{p} + \sigma_\epsilon^2\mathbf{I}$. The correlation matrix $\mathbf{R}_\mathrm{po}$ is an $N \times m$ matrix, and its (i,j)-th elements is defined as in $R(\boldsymbol{\nu}, \mathbf{n}_i, \mathbf{x}_j)$; $\mathbf{R}_\mathrm{p}$ is an $N \times N$ correlation matrix, and its (i,j)-th elements is $R(\boldsymbol{\nu}, \mathbf{n}_i, \mathbf{n}_j)$.

According to the conditional distribution theorem for multivariate normal distribution (Appendix A), we have

$$(\mathbf{y}_\mathrm{p}|\mathbf{y}_\mathrm{o}, \boldsymbol{\theta}) \sim N(\mathbf{g}_\mathrm{p}(\mathbf{X}_\mathrm{p}, \boldsymbol{\beta}) + \mathbf{\Sigma}_\mathrm{po}\mathbf{\Sigma}_\mathrm{o}^{-1}(\mathbf{y}_\mathrm{o} - \mathbf{g}_\mathrm{o}(\mathbf{X}_\mathrm{x}, \boldsymbol{\beta})), \mathbf{\Sigma}_\mathrm{o} - \mathbf{\Sigma}_\mathrm{po}\mathbf{\Sigma}_\mathrm{o}^{-1}\mathbf{\Sigma}_\mathrm{po}^T). \quad (3.10)$$

Equation (3.10) provides the predictive distribution of the discretized geometric feature based on the observed metrology measurements $\mathbf{y}_\mathrm{o}$. The negative amount $\mathbf{\Sigma}_\mathrm{po}\mathbf{\Sigma}_\mathrm{o}^{-1}\mathbf{\Sigma}_\mathrm{po}^T$ in the variance term comes from the observed measurements $\mathbf{y}_\mathrm{o}$ and their correlations with the predicted locations. Basically, the information in the observed metrology data helps reduce the uncertainty in predicting the geometric feature. The more metrology data are observed or the stronger the correlations between $\{\mathbf{n}_i\}_{i=1:N}$ and $\{\mathbf{x}_i\}_{i=1:m}$ are, the less uncertainty remains in the predictive distribution of the geometric feature. In practice, the parameter $\boldsymbol{\theta} = (\boldsymbol{\beta}, \kappa^2, \sigma_\epsilon^2, \boldsymbol{\nu})$ is unknown and will be estimated using the observed metrology data. A common treatment is to plug the MLE estimate $\hat{\boldsymbol{\theta}}$ into Distribution (3.10). The predictive distribution with the plugged-in parameter estimates looks like:

$$(\hat{\mathbf{y}}_\mathrm{p}|\mathbf{y}_\mathrm{o}, \hat{\boldsymbol{\theta}}) \sim N(\mathbf{g}_\mathrm{p}(\mathbf{X}_\mathrm{p}, \hat{\boldsymbol{\beta}}) + \hat{\mathbf{\Sigma}}_\mathrm{po}\hat{\mathbf{\Sigma}}_\mathrm{o}^{-1}(\mathbf{y}_\mathrm{o} - \mathbf{g}_\mathrm{o}(\mathbf{X}_\mathrm{x}, \hat{\boldsymbol{\beta}})), \hat{\mathbf{\Sigma}}_\mathrm{o} - \hat{\mathbf{\Sigma}}_\mathrm{po}\hat{\mathbf{\Sigma}}_\mathrm{o}^{-1}\hat{\mathbf{\Sigma}}_\mathrm{po}^T), \quad (3.11)$$

where $\hat{\mathbf{\Sigma}}_\mathrm{o} = \hat{\kappa}^2\hat{\mathbf{R}}_\mathrm{o} + \sigma_\epsilon^2\mathbf{I}$, $\hat{\mathbf{\Sigma}}_\mathrm{p} = \hat{\kappa}^2\hat{\mathbf{R}}_\mathrm{p} + \hat{\sigma}_\epsilon^2\mathbf{I}$, $\hat{\mathbf{\Sigma}}_\mathrm{po} = \hat{\kappa}^2\hat{\mathbf{R}}_\mathrm{po} + \hat{\sigma}_\epsilon\mathbf{E}$. The matrixes $\hat{R}_\mathrm{o}$, $\hat{R}_\mathrm{o}$ and $\hat{R}_\mathrm{po}$ are the correlation matrixes with the value $\hat{\boldsymbol{\nu}}$ plugged in.

We can reconstruct the geometric feature by drawing a sample from the multivariate distribution specified in Equation (3.11). The sample will be one realization

of the discretized geometric feature. The density of prediction locations is supposed to be much higher than the actual metrology measurements, i.e., $N \gg m$. The measured sites $\{\mathbf{x}_i\}_{i=1:m}$ is usually a subset of the prediction sites $\{\mathbf{n}_i\}_{i=1:N}$. That allows the reconstructed surface to provide a closer representation of the geometric feature than the handful of metrology observations scattered over the surface. Once a surface is predicted or reconstructed, the form error of the feature, denoted by h, is estimated by finding the maximum inscribing and the minimum circumscribing geometries that bound all points of the predicted surface. This treatment follows the Taylor's principle [15]. To account for the uncertainty in data and model, we need to repeat the above procedure T times. T needs to be a big number to ensure a good approximation, and we use $T = 10,000$ for the studies in Section III.F. Fig. 13 (a) shows a 95% predictive band of the predicted surface and the average predicted surface, together with the measured and the true values of the surface. This is a simulated case, so we know the true surface.

Finally, we have T estimates of the form error, $\hat{h}_1, \hat{h}_2, \ldots, \hat{h}_T$. Each of them is calculated by applying the Taylor's principle to an individual predicted surface. Using the T estimates of the form error, we can have an empirical distribution of h, e.g., the histogram in Fig. 13 (b). This histogram is an empirical predictive distribution of the form error, given the metrology measurements $\mathbf{y}_o$ and the GP model. When a large set of metrology data is used, the distribution of h is centered around the actual form error. So it makes sense to use the median of the empirical distribution of h, denoted by $\hat{h}_{(0.5)}$, as the final estimate of the form error.

In fact, the predictive distribution of h contains richer information than a point estimate of the form error. It allows engineers to quantify the decision risk on part acceptance. In practice when the sample size is limited, one may want to be more conservative in accepting the part. For example, instead of using the median $\hat{h}_{(0.5)}$,

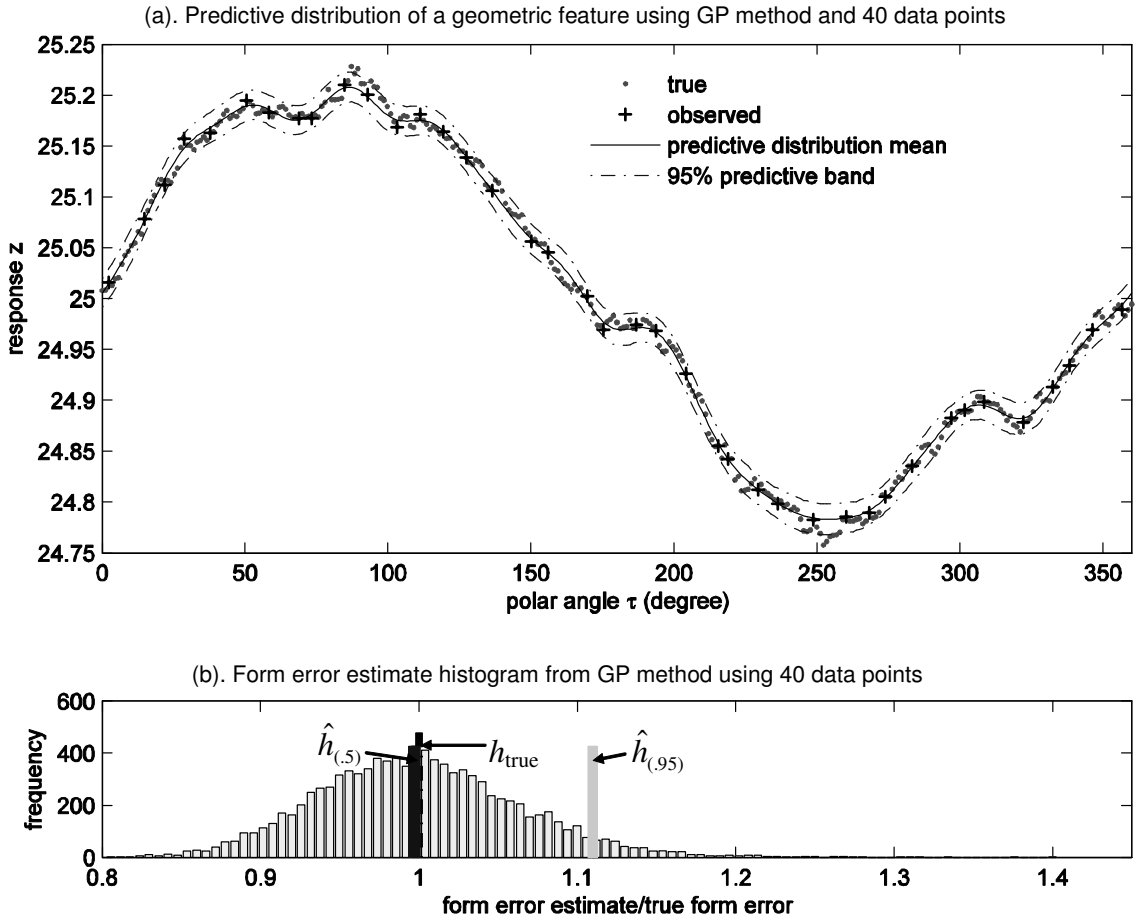

Fig. 13. Predictive distribution of roundness feature and form error estimate histogram

one could use the 95 percentile of the empirical distribution, $\hat{h}_{(0.95)}$, to compare with a pre-determined form tolerance. If $\hat{h}_{(0.95)}$ is smaller than the pre-determined form tolerance, one is more than 95% confident that the part's form error is less than the form tolerance. Which exact percentile to use can be adjusted based on the functionality requirements and the cost of making a wrong decision.

F. Validate Proposed Model and Compare with Traditional Methods

This section compares the GP method for form error estimate with the two traditional methods: the MZ and the OLS methods. We choose these two methods as the reference for comparison primarily because they are still the most popular ones in the literature and are widely used in form error assessment software provided by CMM manufacturers.

In this section, we focus on comparing the unbiasness of the form error estimates from the three different methods. We calculate the ratios of the estimated form errors over the true form errors. The true form errors are known for the simulation studies. If the calculated ratio is closer to 1, the estimate is less biased. We do not compare the probabilistic decision-making procedure allowed by the GP method, which produces a predictive distribution of the form error h, with the MZ and OLS methods. This is because the MZ and OLS methods do not take the uncertainty information into account in their decision making procedure. But we believe that it is an advantage of the GP method to provide the distribution information, which can quantify the decision risk for industrial practices.

We do not compare the speed of these three methods because the GP method uses a large number of replications to get the predictive distribution of the form error, while the other two do not. The OLS method only calculates a point estimate and

is faster. Finding the exact MZ solution could be computationally demanding, but many fast approximation algorithms are available. In the dissertation, we actually use the minimax estimate to approximate the MZ. The actual computation time of the GP method ranges from tens of seconds to several minutes, which should be acceptable to practitioners.

Section III.F.1 performs the comparison using a set of simulated data. We simulate three different manufacturing scenarios for each of the following two geometric features: straightness and roundness. For each scenario, different sizes of samples, ranging from eight to eighty for straightness and for roundness, are used to estimate the form error. The sample size, denoted by m, refers to the number of the measurements taken from different locations on a part.

In Section III.F.2, we use a CMM to obtain actual coordinate measurements from two parts with a straightness feature and a round feature, respectively. We apply the GP method, the MZ method, and the OLS method to estimate the form error and other parameters for each of the two features.

1. Comparison using simulated data

We follow the following procedures to implement the simulation study for each manufacturing scenario in this section (A similar procedure was used by [14] to compare the performance of the MZ method with the OLS method.):

(i). Simulate one single geometric feature.

(ii). Generate a dense enough set of measurements on the selected geometric feature so that the measurements represent the actual geometry well. In the simulation, we take a measurement every 0.5 millimeters. This density is considered dense enough by practitioners. Then we get a total of N points, $N = L/0.5$, where L is the length of a straightness feature or the circumference of a roundness feature. Determine the

form error from the N points using the MZ method, and treat it as the "true" form error.

(iii). Select m data from the set of dense measurements, and treat them as the observed metrology measurements of the geometric feature. The m locations and their corresponding observations are chosen using a maximin distance Latin Hypercube sampling, a method that has the m samples evenly spread over the feature space. For more details on the maximin distance Latin Hypercube sampling, please refer to Reference [47] (Page 150).

(iv). For $m = 8, 10, 15, 20, 30, 40, 80$, determine the form error estimate, and denote it by $\hat{h}_{\text{OLS}}(m)$ when using the OLS method, by $\hat{h}_{\text{MZ}}(m)$ when using the MZ method, and by $\hat{h}_{\text{GP}}(m)$ when using the GP method. The GP method provides a distribution for estimating form errors while the MZ and OLS methods provides only point estimates. So we need to pick a point estimate from the form error distribution such that we can compare these three methods' performances. Here we choose $\hat{h}_{(0.5)}$, the median of the distribution of the form error (defined in Section III.E.2), because it centers around the true form error when a relative large number of metrology data are available.

(v). Calculate the estimate ratios, $\hat{h}_{\text{OLS}}(m)/h^*$, $\hat{h}_{\text{MZ}}(m)/h^*$, and $\hat{h}_{\text{GP}}(m)/h^*$. Recall that h^* is the true form error. A ratio closer to 1 indicates a less biased estimate.

(vi). Repeat Step (iii) to (v) 50 times for each m, and generate a box-whisker plot of the estimate ratios calculated in Step (v).

a. Straightness feature

In this subsection, we simulate the straightness feature. Manufacturing errors of a straightness feature usually include surface deflection, waviness, and random error. Depending on what manufacturing process is used to produce the feature, one of

the errors could dominate in the measurements. For instance, when using a lathe to machine the feature (i.e., a turning process), surface deflection could be more prominent than other types of errors because of the force exerted perpendicularly to the surface. We simulate different scenarios using the generating function from [14] as follows:

$$v = v_0 + \varrho x - \frac{64}{L^6} R(x^3(L-x)^2) + A\sin(\frac{2\pi}{\lambda}x) + \epsilon, \tag{3.12}$$

where the first two terms $v_0 + \varrho x$ represents the rigid-body motion during a fixturing process, the third term represents the surface deflection, the fourth term is a wave pattern, and the last term is the i.i.d. random error assumed to be $N(0, \sigma_\epsilon^2)$. Including an i.i.d. pure random error term sometimes creates outliers jumping out of the geometric surface. The existence of such outliers creates abrupt discontinuity in the geometry and may not reflect the actual surface well. To alleviate the discontinuity problem, we use a three-point moving average window to smooth the i.i.d. random errors. The meanings of the other parameters in Equation (3.12) are as follows: L is the length of the straightness feature; A is the sine wave amplitude; λ is the wavelength; R is the deflection range. We simulate a feature of length $L = 200$ mm. Table III shows the three simulated manufacturing scenarios, corresponding to a milling, a turning, and a grinding process, respectively. The parameters in Table III are determined from the typical process capability associated with each of the manufacturing processes [26].

b. Roundness feature

The dissertation adopts the roundness feature generator from [13] as follows:

$$u = u_0 + (r + A_1\sin(4\tau) + A_2\cos(3\tau) + A_3\sin(7\tau) + A_4\cos(10\tau) + \epsilon)\cos\tau \tag{3.13}$$

$$v = v_0 + (r + A_1\sin(4\tau) + A_2\cos(3\tau) + A_3\sin(7\tau) + A_4\cos(10\tau) + \epsilon)\sin\tau \tag{3.14}$$

Table III. Manufacturing scenarios for a straightness feature

	Process characteristics	δ (mm)	ϕ	A (mm)	λ (mm)	R (mm)	σ_ϵ (mm)
Case I	Sine wave dominates (face milling)	.04	.02	.03	20	.015	.017
Case II	Deflection dominates (turning)	.05	.01	.005	10	.025	.009
Case III	Random errors dominate (grinding)	.03	.01	0	N/A	0	.003

where u_0 and v_0 are the origin of the roundness feature, r is the radius, τ is the polar angle, $A_1 \sin(4\tau) + A_2 \cos(3\tau) + A_3 \sin(7\tau) + A_4 \cos(10\tau)$ represents the systematic error, and other notations are the same as defined before. Table IV summarizes three different manufacturing scenarios.

Table IV. Manufacturing scenarios for a roundness feature (unit: mm)

	Process characteristics	u_0	v_0	A_1	A_2	A_3	A_4	r	σ_ϵ
Case I	Three-lobed systematic errors dominate and radius change (turning)	.2	.02	N/A	.03	N/A	N/A	25.03	.01
Case II	General systematic errors dominate (turning)	.03	.2	.002	-.015	-.01	-.008	25	.012
Case III	Random errors dominate (turning)	.01	.15	N/A	N/A	N/A	0	25	.017

c. Results and discussion

Figs. 14-16 show the results of form error estimation for the straightness feature, and Figs. 17-19 show the estimation results for the roundness feature. In each box-whisker plot, the locations of the upper limit, the 75% quantile, the median, the 25% quantile and the lower limit are shown. The crosses outside of the upper and lower limits are

usually considered as "outliers". The dashed line indicates that the estimate of the form error is the same as the true form error, i.e., the estimated ratio equals to one. In other words, the best method is the one that consistently produces box-whisker plots closest to the dashed line.

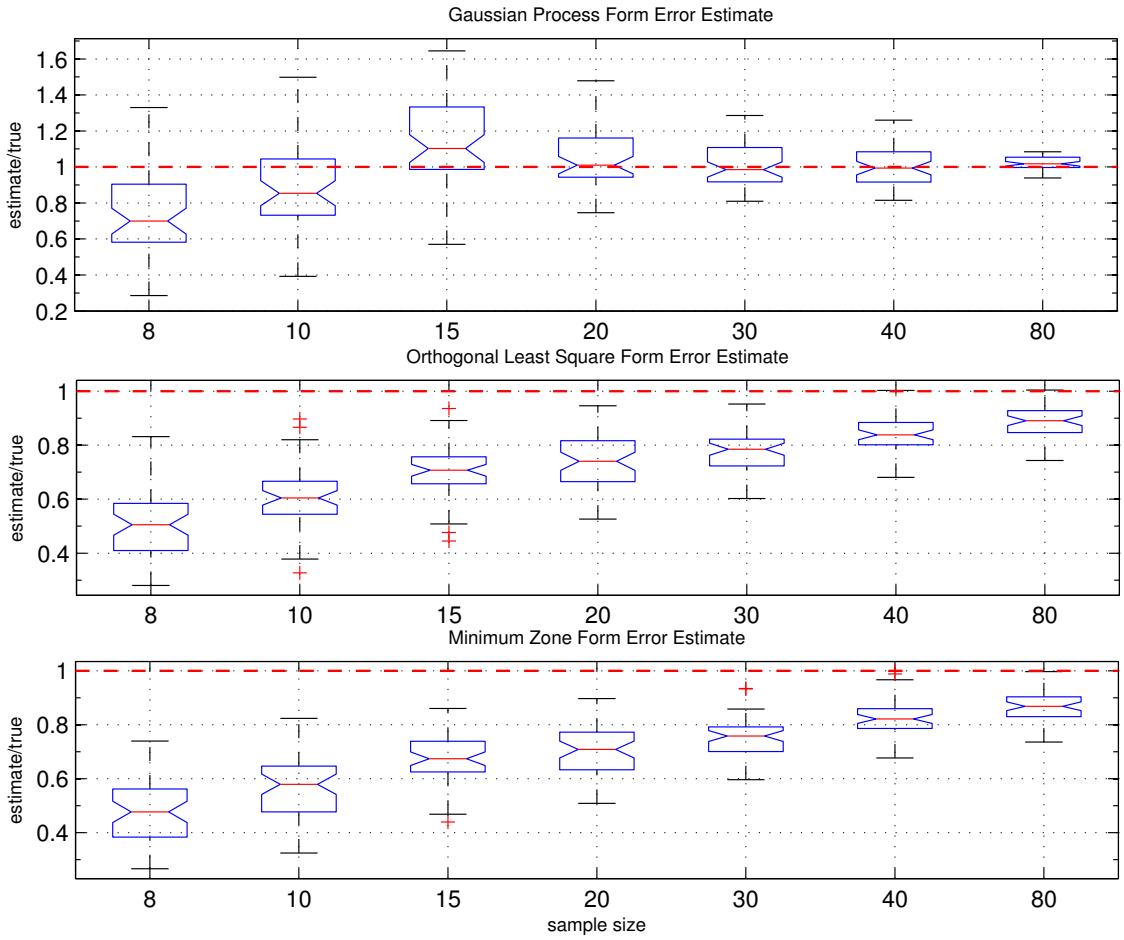

Fig. 14. Form error estimate comparison for straightness Case I

From Figs. 14-19, we observe the following:

(i). The proposed GP method performs significantly better than the OLS and MZ methods when systematic manufacturing errors exist. When the sample size grows larger, the GP method tends to be unbiased. This appears to confirm what we observe

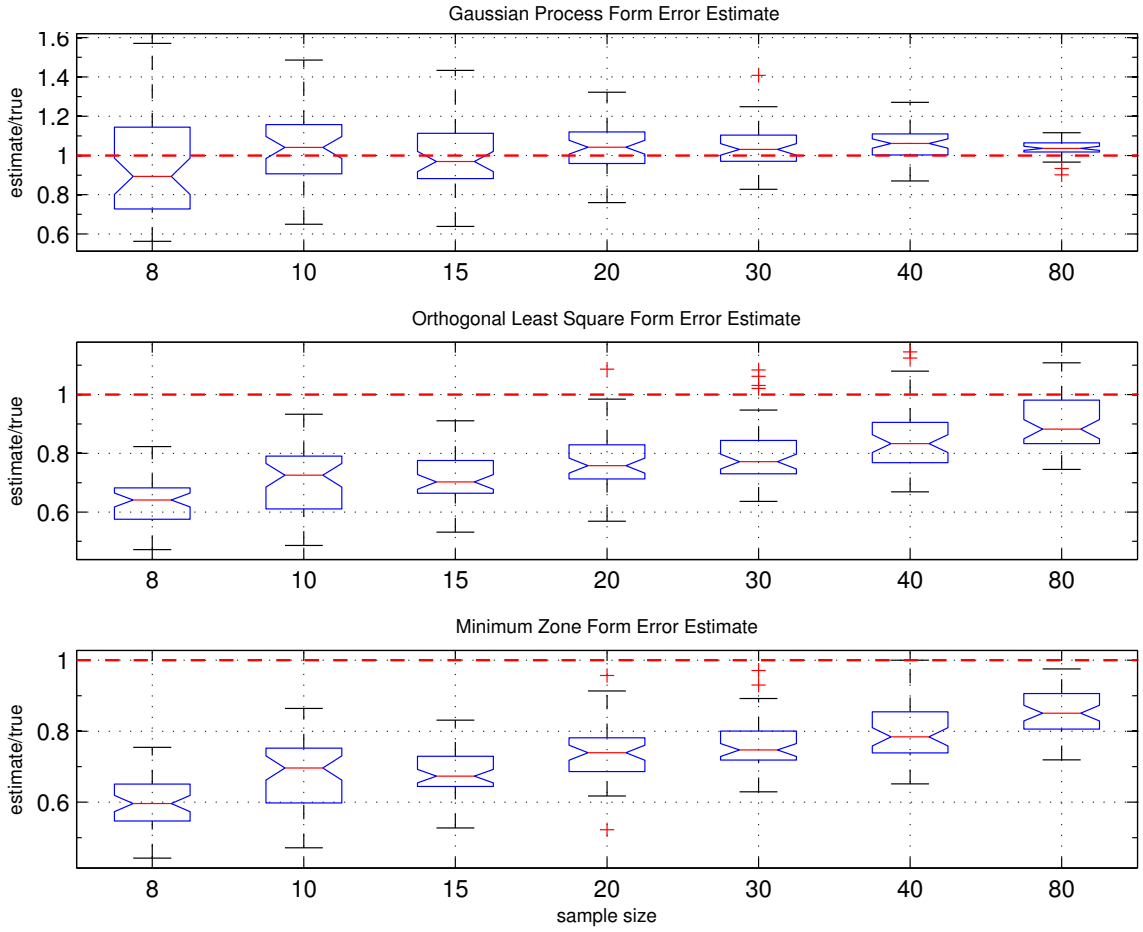

Fig. 15. Form error estimate comparison for straightness Case II

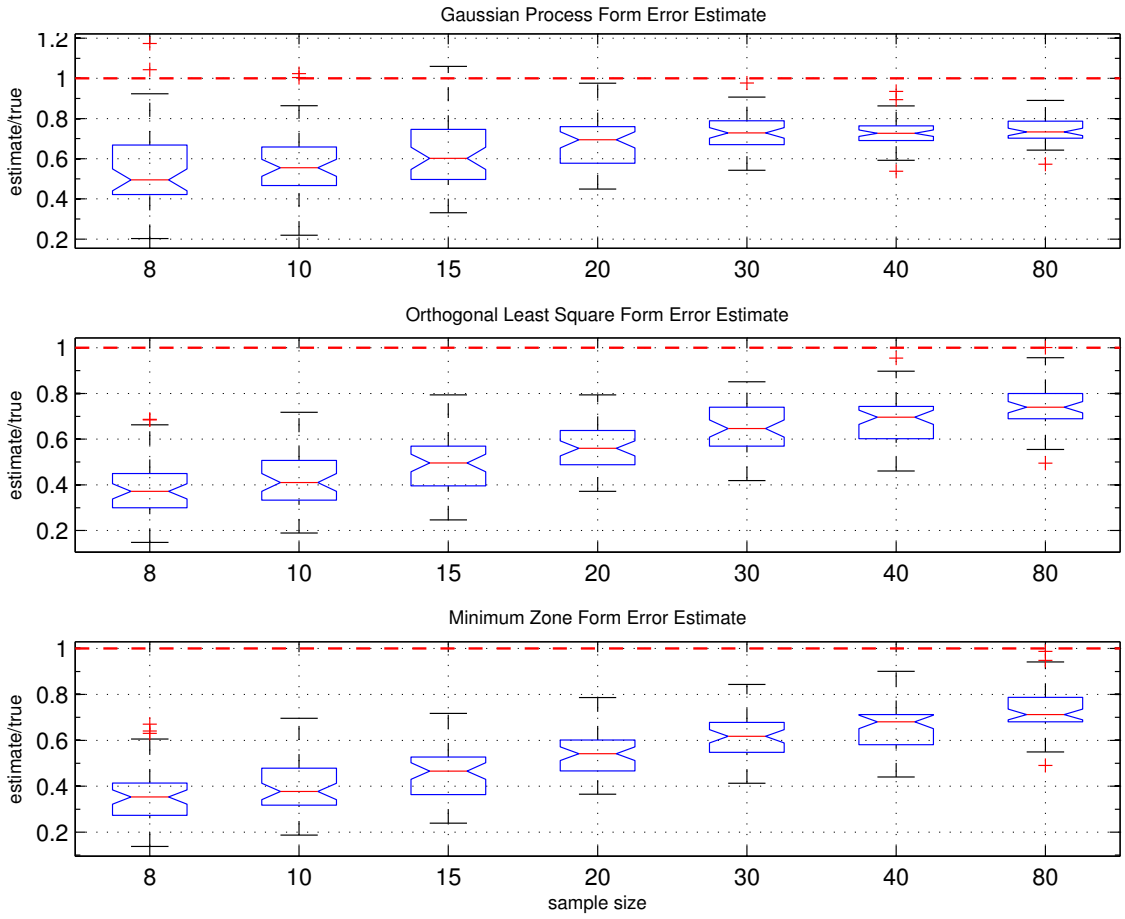

Fig. 16. Form error estimate comparison for straightness Case III

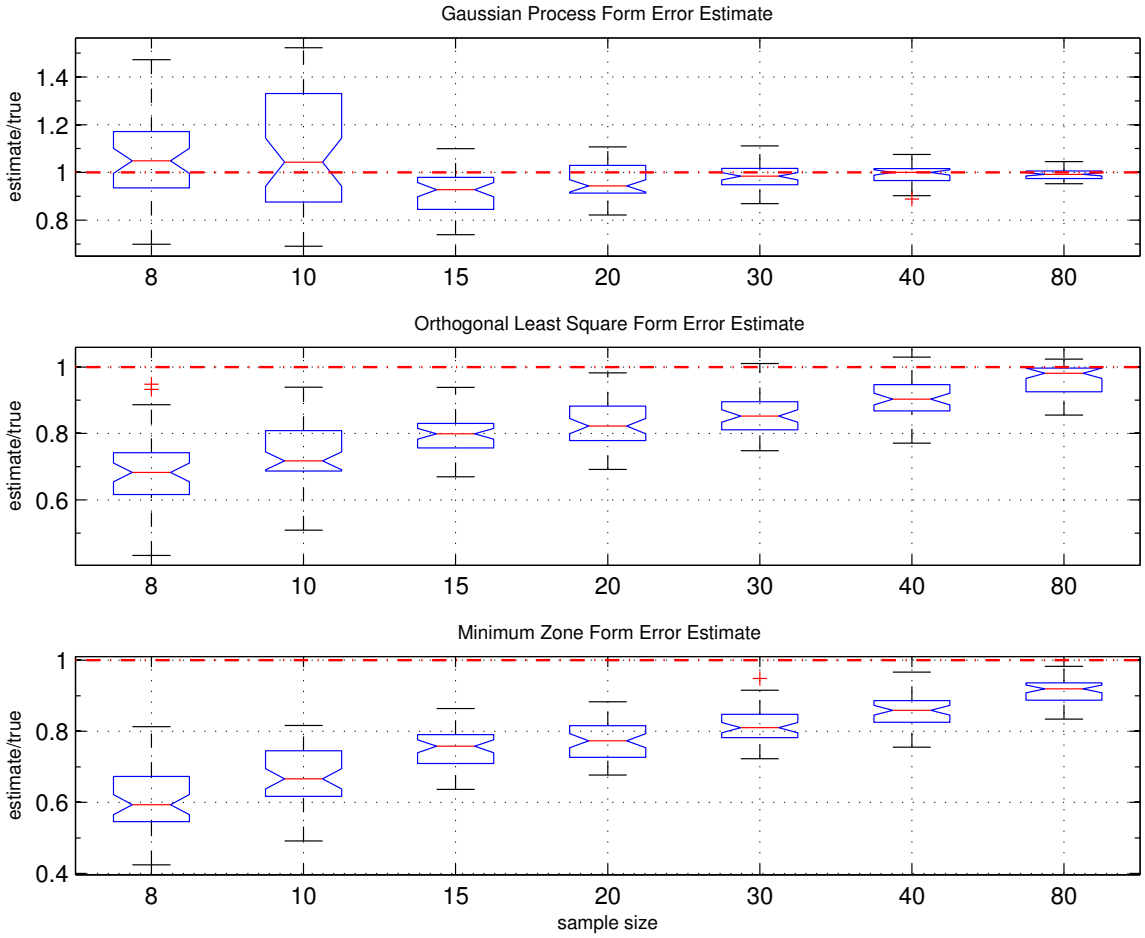

Fig. 17. Form error estimation comparison for roundness Case I

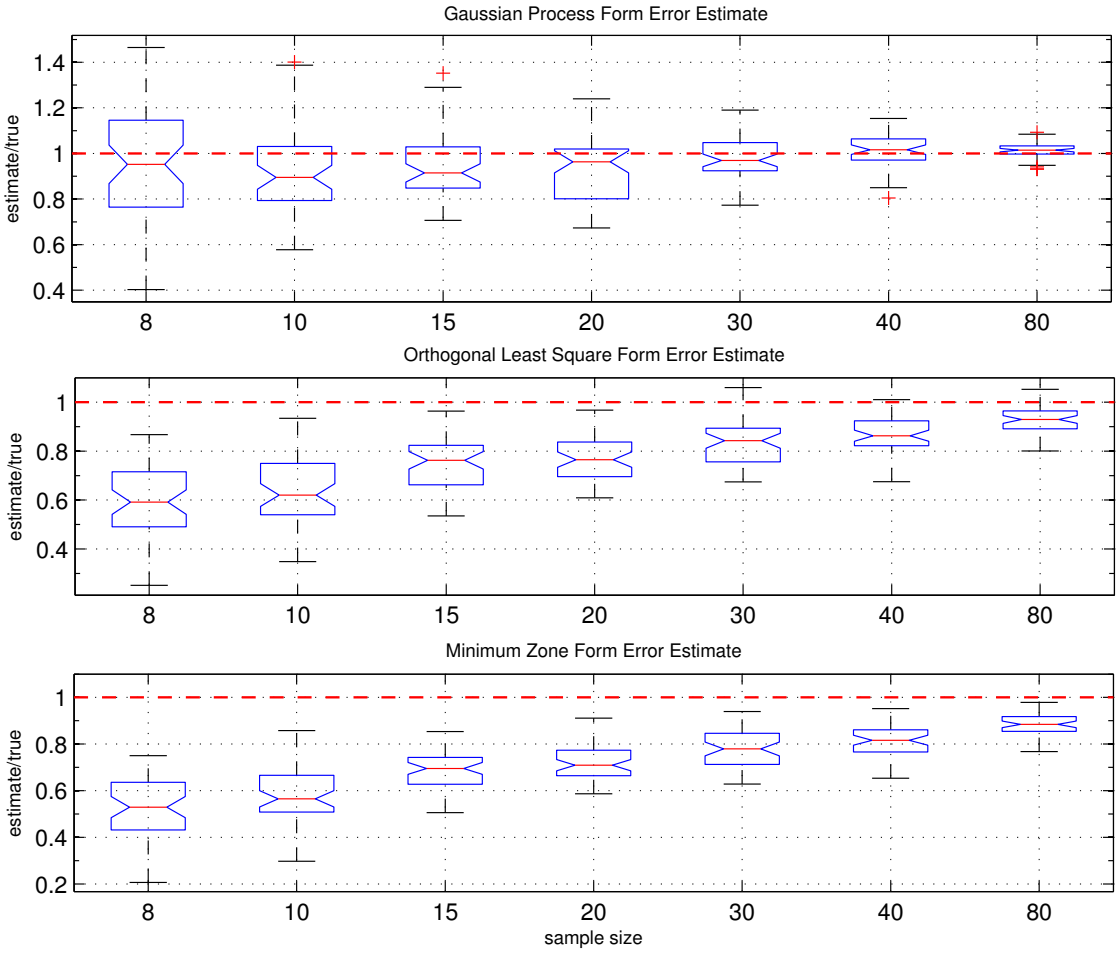

Fig. 18. Form error estimation comparison for roundness Case II

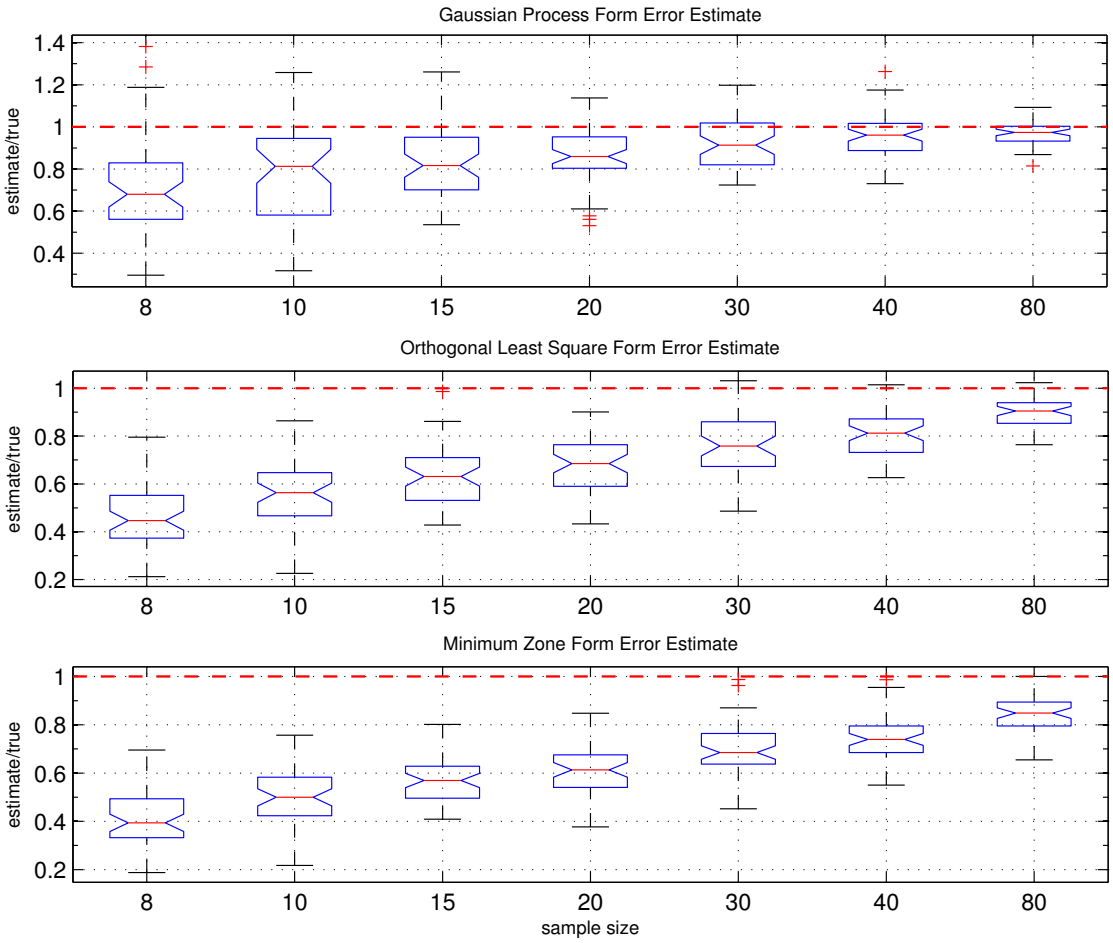

Fig. 19. Form error estimation comparison for roundness Case III

in the empirical distribution of form error estimate in Fig. 13 (b). By comparison, the MZ and OLS methods tend to underestimate the form error, even when using a relatively large sample size, e.g., $m = 40$ or 80. Reference [14] also mentioned the underestimation of the MZ and OLS methods for straightness features. We believe that our form error estimation/assessment benefits from the GP method's ability to capture the systematic manufacturing errors, while the other two methods treat the handful of metrology measurements as a complete representation of the entire feature. Please refer to Fig. 13 (a) which shows the predicted surface for a roundness feature.

(ii). In the cases when only random manufacturing errors exist, i.e., Fig. 17 and 19, the GP method performs similarly to the OLS method and slightly better than the MZ method. This is expected since the OLS method assumes that the error term is i.i.d. random noises. When the systematic error term vanishes, the GP model is essentially the same as the OLS model.

(iii). The GP method suffers from not having sufficient information when the metrology data sample is small as the other two methods do, but to a less degree. Insufficient information from small sample generally leads to wider predictive distributions, meaning more uncertainty.

(iv). In the simulations, we decide the smallest sample size according to the number of unknown parameters used in the GP model, which is five for the straightness and six for the roundness. We start with a sample size roughly 1.5 times the number of unknown parameters. Prior literature has not yet settled on how many samples will result in good predictions. Reference [2] suggested using three observations per parameter as a rule of thumb for good model performance. That translates to a sample size of 15 and 18, respectively, for the two features that the dissertation studies. From the simulation results, we notice that the GP method produces reasonably good estimate of form error when the sample size is larger than 15 for straightness or

larger than 20 for roundness. Citing the work [57], Reference [15] pointed out that a sample size of 10 to 20 is usually needed to estimate the error and the parameters well. What the simulation suggests appears consistent with the recommendation of Reference [57].

(v). In the cases where the number of the metrology data is fewer than 1.5 times the number of unknown parameters, the GP method may not produce accurate enough predictions, so the alternative method should be used.

2. Form error estimation using actual CMM measurements

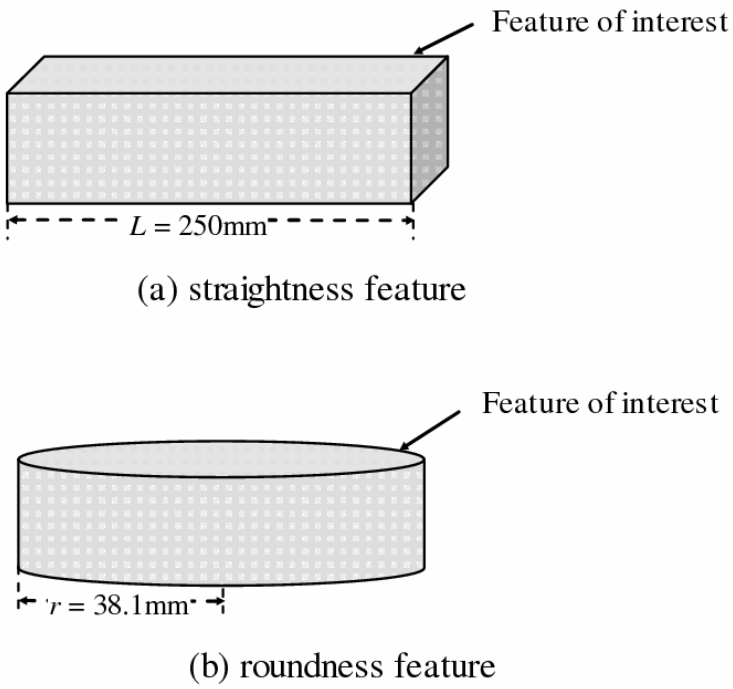

Fig. 20. Sketches of two real parts

We use a CMM to obtain coordinate measurements from two parts. The first one is a straight block with 250 millimeters in length, which is shown in Fig. 20 (a) and is

manufactured by a face milling process. The other one is a cylinder with a roundness feature of 38.1 millimeters in radius, which is shown in Fig. 20 (a) and is manufactured by a rough turning process. We take a set of dense metrology measurements (i.e., one measurement every .5 millimeters) for both the straightness feature and the roundness feature. As a result, we have a total of 500 points taken from the straightness feature, and 480 points taken from the circle. The Taylor's principle is applied to the dense metrology measurements, and the calculated form error is treated as the true form error h^*. For the straightness feature, the form error $h^* = .052$ mm; for the roundness feature, $h^* = .43$ mm.

Then we take a smaller, more practical number m, of CMM observations from each part to estimate the form error. Again, the three methods, the GP, OLS, and MZ methods, are used. The sample sizes we used are 15, 20 and 30. Figs. 21-22 summarize the comparison results for the straightness feature and the roundness feature, respectively. Consistent with the previous simulation results, the GP method-based form error estimate performs much better than the other two estimates and appears to be less biased.

In addition, we obtain the estimation of the dimension parameter (only for the roundness), and the variance of the random errors. Table V shows the estimation results and gives the mean values and the associated standard deviations (in bracket).

For the roundness feature, manufacturing engineers are interested in estimating of the radius r and using it for dimensional quality control. For all different sample sizes, the estimates from the GP method and from the OLS method are very close in terms of both the average estimates and the standard deviations. The estimate $\hat{r}$ from using the MZ method deviates more noticeably from the estimates from the OLS and GP methods, and the standard deviation of the MZ estimate is about three

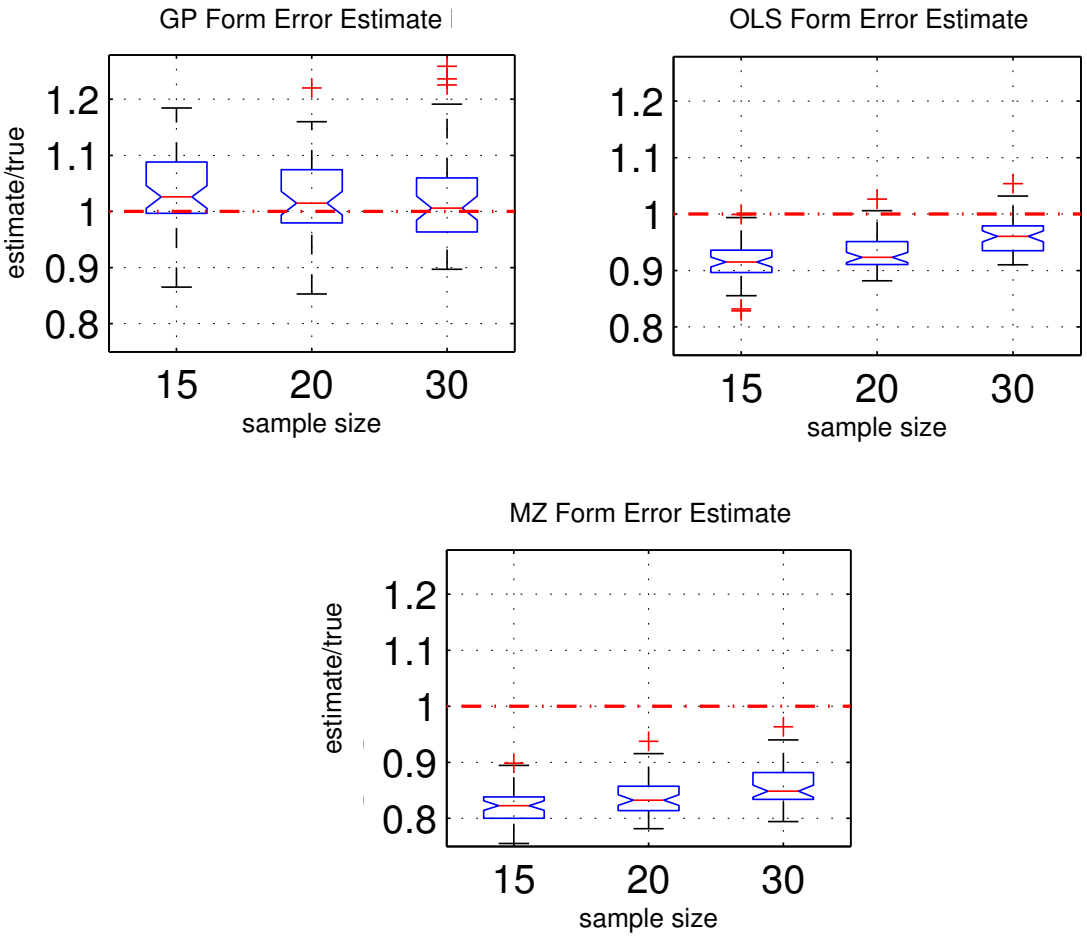

Fig. 21. Form error estimate comparison using real metrology measurements for straightness

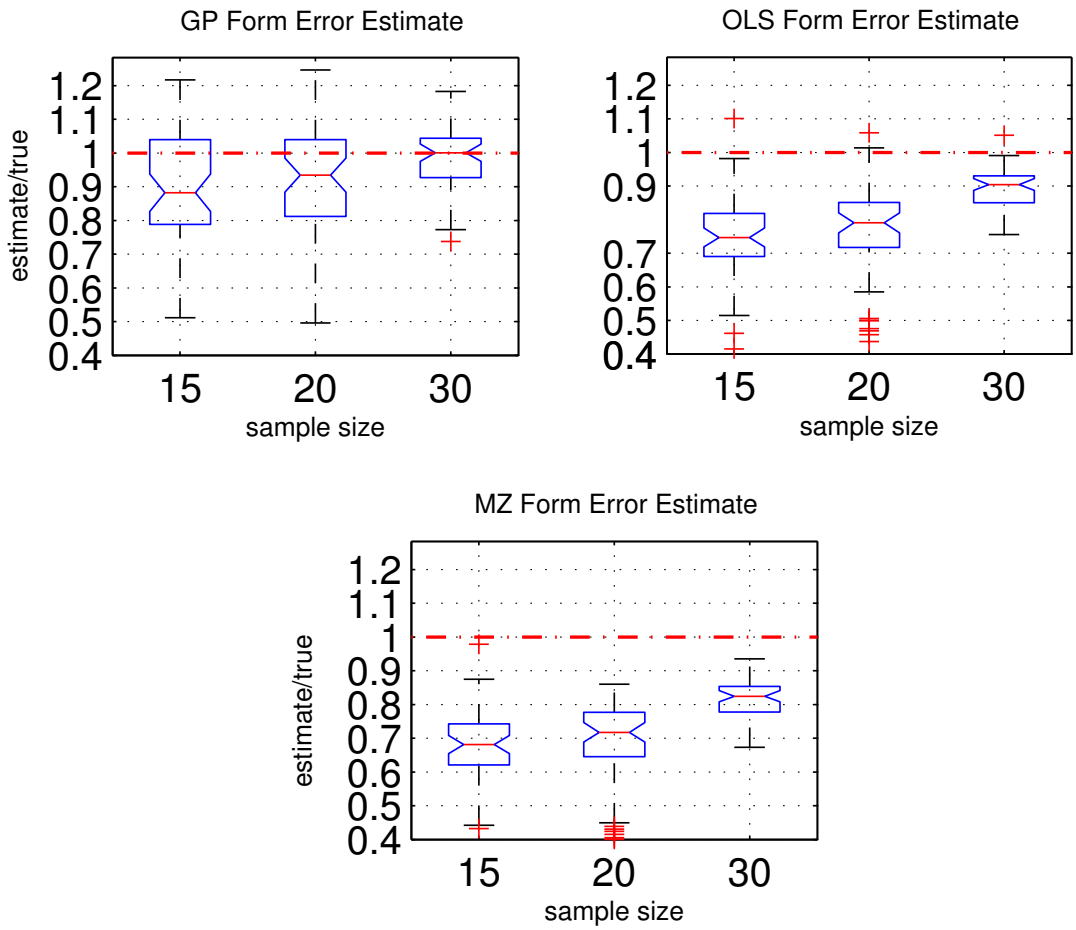

Fig. 22. Form error estimate comparison using real metrology measurements for roundness

Table V. Parameter estimate comparison

	Straightness		Roundness				
sample size	$\hat{\sigma}_\epsilon^{\mathrm{OLS}}$ (μ m)	$\hat{\sigma}_\epsilon^{\mathrm{GP}}$ (μ m)	$\hat{\sigma}_\epsilon^{\mathrm{OLS}}$ (μ m)	$\hat{\sigma}_\epsilon^{\mathrm{GP}}$ (μ m)	$\hat{r}_{\mathrm{OLS}}$ (10^{-2}mm)	$\hat{r}_{\mathrm{MZ}}$ (10^{-2}mm)	$\hat{r}_{\mathrm{GP}}$ (10^{-2}mm)
15	15 (.65)	3.4 (1.1)	93 (18)	21 (12)	3810 (1.1)	3807 (3.3)	3809 (1.2)
20	14 (.61)	3.4 (1.2)	87 (16)	22 (11)	3810 (1.3)	3806 (3.4)	38.10 (1.3)
30	14 (0.48)	3.3 (1.2)	89 (7.4)	18 (9.8)	3810 (.71)	3804 (2.6)	3810 (.61)

times larger than those of the OLS and GP estimates. This suggests that using the MZ method for parameter estimation might not be a good practice. The estimated standard deviation of random error $\hat{\sigma}_\epsilon$ can be obtained by the GP method and the OLS method but not by the MZ method. In fact, every decision in a MZ procedure is deterministic. One may notice that the GP method yields a smaller estimate of random error $\hat{\sigma}_\epsilon$ than the OLS method. This is because the OLS method treats both the errors, ξ and ϵ, as the random error and thus may inflate the estimate $\hat{\sigma}_\epsilon$. The inflation in the estimate $\hat{\sigma}_\epsilon$ will cause a loss of power in subsequent inferences [35].

Other parameters such as ϱ, u_0 and v_0 can be estimated as well. These parameters are related to the rigid body motion of the part during a fixturing process. They are not directly involved in the form error assessment but only work as a compensation of imperfect alignment. Their estimated values are indeed small because of the soft-fixturing procedure used in a CMM measuring process. To save space, we do not list them here.

G. Chapter Summary

This chapter presents a GP method for form error assessment. The comparisons show the GP method generally gives a less biased estimate of the form error than the traditional MZ and OLS methods. The simulation results indicate that a sample of 15 or more metrology data should be used with the proposed GP method, at least for the two geometric features under consideration. This result is consistent with the recommendation from previous studies [57]. The GP method produces a predictive distribution of the form error, allowing a decision-maker to take into account the uncertainty from the modeling and the sampling.

An additional note is on the random error term in the GP model. Reference [15] mentioned that the rule of thumb in practice is that one may ignore the measurement error if the tolerance size is 10 times larger than the CMM's uncertainty. The CMM we used is a Sheffield Discovery II D-8. Its calibrated volumetric accuracy and repeatability are 4.7 and 1.66 microns in range, respectively. The combined uncertainty is around 6 microns. Compared with the form errors of 0.052 millimeters for straightness and 0.43 millimeters for roundness, the above rule of thumb certainly holds for the roundness feature and approximately holds for the straightness feature (about 8 times). So our treatment of attributing all the random error to the manufacturing process is reasonable for the products we analyzed. However, some high-precision manufacturing processes, such as grinding, lapping and honing, have a typical tolerance limit of a few microns (e.g., 8 microns for grinding). Then, the above rule-of-the-thumb will not be strictly satisfied. Therefore in general, engineers need to consider, and eventually to eliminate, the influence of the measurement error in order to reduce the false positives in the process of quality assurance. The current GP model can be extended to include the measurement error, but it requires

repeated measurements to enable differentiating measurement errors from random manufacturing errors.

CHAPTER IV

BAYESIAN HIERARCHICAL MODEL FOR COMBINING TWO-RESOLUTION METROLOGY DATA

This dissertation proposes a Bayesian predictive method that combines the metrology datasets from two different resolutions when the matching and alignment between the two datasets is unknown. We devise a hierarchical predictive model as such: at the lower level, a Gaussian process (GP) model developed in Chapter III is used for representing the low-resolution data; and at the upper level, a *neighborhood linkage model* is established to link each high-resolution output as a kernel regression of all the low-resolution information in its neighborhood. Different from the linkage model used in the computer experiments [40], the proposed linkage model, conditioned on a match between the two-resolution datasets, extends the one-to-one linkage to a one-to-many linkage. The reason we make this extension is, despite all the alignment efforts, there is no guarantee to find the true match with the presence of distortions [36], [54].

We develope a heuristic matching algorithm, which outputs several most probable match candidates. For each match candidate, an optimization minimizes the sum of the squared Euclidean distances between matched data points to find the corresponding best transformation. Conditioned on each match and its corresponding transformation, the hierarchical predictive model could predict the part's surface using the two-resolution metrology data. Furthermore, we conduct a Bayesian model averaging of the predictive models over the pool of candidate matches found by the heuristic matching. This Bayesian model averaging procedure assigns weights to different matches according to how well they support the observed data, and then produces the final combined predictions of the entire surface. Fig. 23 summarizes the overall framework of the proposed method. We will discuss the individual components

in more detail in the following sections.

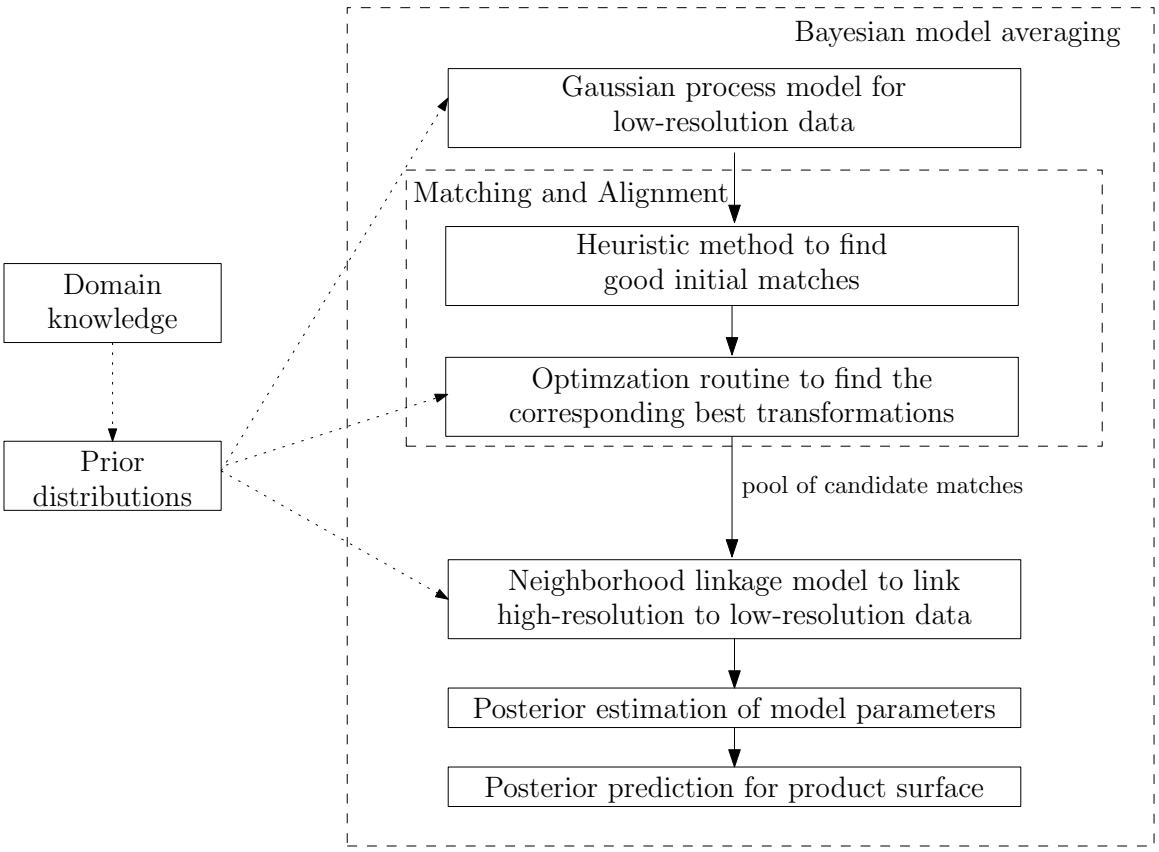

Fig. 23. Overall framework of the proposed Bayesian hierarchical model

A. Low-resolution Data Model

Chapter III presents a GP model for single-resolution metrology data of a manufactured part. This GP model is adopted for the low-resolution metrology data in this Bayesian hierarchial model to combine two-resolution metrology data.

The GP model for low-resolution data is as follows:

$$y_l(\mathbf{x}_i) = \eta_l(\mathbf{x}_i) + \epsilon_l, \quad i = 1, \ldots, m_l. \tag{4.1}$$

Here a subscript "l" is associated with each term because the model is used here for the low-resolution data, although the same model can be used for the high-resolution data as well. In Equation (4.1), $y_l(\mathbf{x}_i)$ is the i-th low-resolution observation, m_l is the number of the low-resolution observations, ϵ_l is the random error, dominated by measurement noises and modeled as i.i.d. $N(0, \sigma_l^2)$, and $\eta_l(\cdot)$ is the low-resolution version of the actual surface. Please note that $\eta_l(\cdot)$ is not exactly the true surface because the low-resolution measurements provide a "blurred" view, rather than a sharp reflection, of the actual surface.

The low-resolution surface $\eta_l(\cdot)$ corresponds to the sum of $\mathbf{f}(\cdot, \boldsymbol{\beta})^T\mathbf{p}_i$ and $\xi(\cdot)$ in Equation (3.5) of Chapter III. So it is a Gaussian process with the i-th mean component being $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta}_l)^T\mathbf{p}_i$ and the covariance function being $\text{cov}(\eta_l(\mathbf{x}_i), \eta_l(\mathbf{x}_j)) = \kappa_l^2 R(\boldsymbol{\nu}, \mathbf{x}_i, \mathbf{x}_j)$, where κ_l^2 and $R(\boldsymbol{\nu}, \mathbf{x}_i, \mathbf{x}_j)$ are the variance and the correlation function, respectively. The three-step procedure to compute the $\mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta}_l)$ for a general geometric feature is provided in Chapter III, Section D. The correlation function $R(\boldsymbol{\nu}, \mathbf{x}_i, \mathbf{x}_j)$ is modeled as a product of Gaussian correlation functions [47]:

$$R(\mathbf{x}_i, \mathbf{x}_j) = \prod_{k=1}^{d} \exp\{-\nu_k (x_{ki} - x_{kj})^2\}, \tag{4.2}$$

where d is the dimension of the input variables, i.e., $\mathbf{x}_i = (x_{1i}, x_{2i}, \ldots, x_{di})$, and $\boldsymbol{\nu}_l = (\nu_1, \nu_2, \ldots, \nu_d)$ are the scale parameters controlling how fast the correlation decays as the between-input distance increases in each dimension. For the metrology applications, $d = 1, 2$, or 3. For this GP model, we summarize the model parameters in $\boldsymbol{\theta}_l = (\boldsymbol{\beta}_l, \kappa_l^2, \sigma_l^2, \boldsymbol{\nu}_l)$.

Denote the location matrix where low-resolution observations are made by $\mathbf{X}_l = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m_l})^T$; denote the low-resolution surface by $\boldsymbol{\eta}_l = (\eta_l(\mathbf{x}_1), \eta_l(\mathbf{x}_2), \ldots, \eta_l(\mathbf{x}_{m_l}))^T$; denote the observed low-resolution data by $\mathbf{y}_l = (y_l(\mathbf{x}_1), y_l(\mathbf{x}_2), \ldots, y_l(\mathbf{x}_{m_l}))^T$. Con-

ditioned on $\boldsymbol{\theta}_l$, the joint distribution of $\boldsymbol{\eta}_l$ and $\mathbf{y}_l$ is as follows:

$$p(\boldsymbol{\eta}_l, \mathbf{y}_l|\boldsymbol{\theta}_l) = N\left(\begin{bmatrix} \mathbf{g}(\mathbf{X}_l, \boldsymbol{\beta}_l) \\ \mathbf{g}(\mathbf{X}_l, \boldsymbol{\beta}_l) \end{bmatrix}, \begin{bmatrix} \kappa_l^2\mathbf{R}_l & \kappa_l^2\mathbf{R}_l \\ \kappa_l^2\mathbf{R}_l^T & \boldsymbol{\Sigma}_l \end{bmatrix}\right), \tag{4.3}$$

where $\mathbf{g}(\mathbf{X}_l, \boldsymbol{\beta}_l)$ is an $m_l \times 1$ column vector whose i-th element is $g(\mathbf{x}_i, \boldsymbol{\beta}_l) = \mathbf{f}(\mathbf{x}_i, \boldsymbol{\beta}_l)^T\mathbf{p}_i$; $\mathbf{R}_l$ is an $m_l \times m_l$ correlation matrix whose (i, j)-th element is defined according to Equation (4.2); and $\boldsymbol{\Sigma}_l = \kappa_l^2\mathbf{R}_l + \sigma_l^2\mathbf{I}$.

According to the conditional distribution theorem for multivariate normal distribution (Appendix A), the distribution of $\boldsymbol{\eta}_l$ conditioned on the observations of $\mathbf{y}_l$ would be:

$$p(\boldsymbol{\eta}_l|\mathbf{y}_l, \boldsymbol{\theta}_l) = N\left(\mathbf{g}(\mathbf{X}_l, \boldsymbol{\beta}_l) - \kappa_l^2\mathbf{R}_l\boldsymbol{\Sigma}_l(\mathbf{y}_l - \mathbf{g}(\mathbf{X}_l, \boldsymbol{\beta}_l)), \kappa_l^2\mathbf{R}_l - \kappa_l^4\mathbf{R}_l\boldsymbol{\Sigma}_l\mathbf{R}_l^T\right). \tag{4.4}$$

This equation indicates that $\boldsymbol{\eta}_l$ can also be considered as a filtered version of the low-resolution data and is free of the random measurement errors. Later, the filtered low-resolution data $\eta_l(\mathbf{x}_i)$, instead of the unfiltered $y_l(\mathbf{x}_i)$, for $i = 1, \ldots, m_l$, will be used when establishing a link to the high-resolution data.

B. Match and Align Two-resolution Metrology Data

Suppose, before projecting onto $\mathbf{p}$, we have a set of high-resolution data, $D_h = \{\mathbf{a}_i^h : i = 1, \ldots, m_h\}$, and a set of low-resolution data, $D_l = \{\mathbf{a}_j^l : \ j = 1, \ldots, m_l\}$, where $m_h << m_l$. Both sets of data points are arbitrarily labeled for identification. The goal is to match a subset of the low-resolution data points to the whole set of high-resolution data points and then find the corresponding transformation.

Matching two datasets refers to establishing the correspondence between two datasets. For example, a found match could be $\mathbf{a}_1^h \to \mathbf{a}_5^l, \mathbf{a}_2^h \to \mathbf{a}_{11}^l, \mathbf{a}_3^h \to \mathbf{a}_{64}^l, \ldots, \mathbf{a}_{20}^h \to \mathbf{a}_{496}^l$, as in Fig. 24 (a). The transformation refers to the rigid body transformation

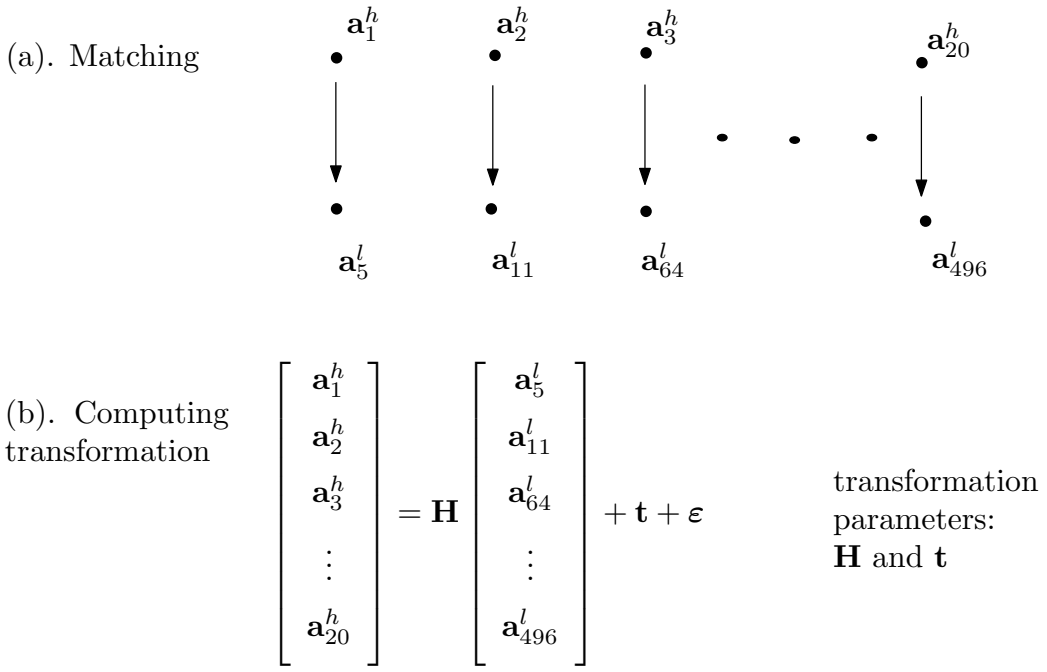

Fig. 24. Two steps to align two-resolution data of sine wave

between the low-resolution dataset and the high-resolution dataset, i.e., the rotation matrix $\mathbf{H}$ and the translation vector $\mathbf{t}$ in Fig. 24 (b). The parameters of the tranformation (i.e., the mapping function) can be computated using the established correspondence. In other words, the alignment procedure normally goes through two steps: first, we find the correspondence between the two datasets, a procedure called matching; second, we computate the transformation parameters based on the found match.

1. Heuristic matching

We develop a heuristic matching algorithm utilizing the invariance property of inter-point distances explained in Section II.D, which was initially introduced for image registration [42]. Mathematically, it means the following: if $(\mathbf{a}_i^h, \mathbf{a}_j^l)$ is a pair of

matching points, then for each other point $\mathbf{a}_k^h$ in D_h ($k \neq i$), there should be a corresponding point, denoted by $\mathbf{a}_s^l$, such that the distances $(\mathbf{a}_i^h, \mathbf{a}_k^h)$ "almost" equals to the distance $(\mathbf{a}_j^l, \mathbf{a}_s^l)$. Here we use "almost" because both measurement datasets are noisy, so an exact equality is difficult to attain. During implementation, this "almost equality" is implemented by setting a threshold for difference allowance. Before doing the heuristic matching, we estimate the nominal surfaces from both datasets and align them together. In this way, the two datasets are very close to each other ensuring that the heuristic algorithm works. The heuristic algorithm proceeds as follows.

Repeat the following two steps for $j = 1, 2, \ldots, m_l$,
(Step 1). Let the first data point $\mathbf{a}_1^h$ in D_h be paired with the j-th data point $\mathbf{a}_j^l$ in D_l.
(Step 2). For $k = 2, 3, \ldots, m_h$, calculate the inter-point distance between $\mathbf{a}_k^h$ and $\mathbf{a}_1^h$, and then find a point in D_l, e.g., the s-th low-resolution point, such that $\parallel \ (\mathbf{a}_s^l - \mathbf{a}_j^l) - (\mathbf{a}_k^h - \mathbf{a}_1^h) \parallel$ is less than some threshold ϖ. If such a low-resolution point is found, $\mathbf{a}_k^h$ and $\mathbf{a}_s^l$ will be considered as a pair of matching points. When there are more than one low-resolution points satisfying the above requirement, select the one giving the smallest difference between $\mathbf{a}_s^l - \mathbf{a}_j^l$ and $\mathbf{a}_k^h - \mathbf{a}_1^h$. Once going through all the m_h high-resolution points, if all of them find their matching points in the low-resolution dataset, we consider a match identified and call the set of the matching pairs a *consistent match*, a term first used in the image registration literature [36]. If not all the m_h high-resolution points find their matching point in the low-resolution data, then we say no consistent match exists for $\mathbf{a}_j^l$, go back to Step 1, and let $\mathbf{a}_1^h$ be paired with the next low-resolution point.

In this algorithm, we basically check each of the m_l cases for consistent matches.

This algorithm is quite general and works with any geometry, even symmetric features such as a circle. Even though m_l is usually a large number, we can still afford to enumerate all the m_l cases for consistent matches. For example, enumerating the $m_l = 1,560$ cases in the sine-wave example takes two minutes, and it is affordable as an off-line operation. The Bayesian model averaging procedure for combining two-resolution data is more computational expensive. So it is more important to eliminate all the inconsistent matches before the Bayesian model averaging.

For features that have clear boundary and are asymmetric, we may utilize some engineering knowledge to further eliminate cases where we will find no consistent matches. For example, we may avoid matching a corner point to some point which lies in the middle of the feature under study. If engineering knowledge can help reduce the number of cases from m_l to a small number, we just need to do Step 1 and 2 above for each of the small number cases.

Suppose the heuristic matching procedure produces K consistent matches. Usually K is much smaller than m_l. We record each consistent match between the two datasets with a matching matrix $\mathbf{M} = (M_{ij})$, where

$$M_{ij} = \begin{cases} 1 & \text{if } \mathbf{a}_j^l \text{ is matched to } \mathbf{a}_i^h, \\ 0 & \text{otherwise.} \end{cases} \tag{4.5}$$

Note $\sum_{i,j} M_{ij} = m_h$. Also, only one low-resolution point will be matched to each high-resolution data point, i.e., $\sum_j M_{ij} = 1$. In this way, the K consistent matches will be denoted as $\{\mathbf{M}_k, k = 1, \ldots, K\}$. We will find their corresponding best transformations and feed them to the Bayesian hierarchical model to combine the two-resolution data. Conditional on each match and its corresponding transformation, the Bayesian hierarchical model makes a set of predictions. All sets of predictions will be combined using Bayesian model averaging over different matches.

Given the two-resolution data, the number of consistent matches K found by the heuristic procedure depends on the distance threshold ϖ. A small ϖ will lead to a fewer number of consistent matches. As a result, the Bayesian hierarchical model will run over a fewer number of matches, and the subsequent inferences will also incur less computation. Having a too small value for ϖ should also be avoided because it would leave out some good matches and undermine the performance of the final combined predictions. For the examples in Chapter V, we set ϖ to be the average inter-point distance in the low-resolution dataset. With this ϖ, for the two examples in Chapter V, the heuristic matching produces $K = 15$ or 30 consistent matches for the circular surfaces and $K = 2$ or 1 consistent matches for the sine-wave surface, respectively.

2. Optimization to calculate transformation

After matching the two datasets, we can calculate the best transformation by solving the following optimization, which minimizes the sum of the squared distances between the corresponding points in the two datasets over possible rigid body transformation:

$$\min_{\substack{t_u,t_v,t_w \\ \rho_u,\rho_v,\rho_w}} \sum_{\substack{i=1:m_h \\ j\in\{j:M_{ij}=1\}}} \| \mathbf{a}_i^h - \left(\mathbf{H}(\rho_u,\rho_v,\rho_w)\mathbf{a}_j^l + (t_u,t_v,t_w)^T\right) \|^2 \tag{4.6}$$

s.t. $\mathbf{H}(\rho_u,\rho_v,\rho_w) =$

$$\begin{bmatrix} \cos(\rho_u)\cos(\rho_v) & \cos(\rho_u)\sin(\rho_v)\sin(\rho_w)-\sin(\rho_u)\cos(\rho_w) & \cos(\rho_u)\sin(\rho_v)\cos(\rho_w)+\sin(\rho_u)\sin(\rho_w) \\ \sin(\rho_u)\cos(\rho_v) & \sin(\rho_u)\sin(\rho_v)\sin(\rho_w)+\cos(\rho_u)\cos(\rho_w) & \sin(\rho_u)\sin(\rho_v)\cos(\rho_w)-\cos(\rho_u)\sin(\rho_w) \\ -\sin(\rho_v) & \cos(\rho_v)\sin(\rho_w) & \cos(\rho_v)\cos(\rho_w) \end{bmatrix};$$

$t_u, t_v, t_w \in \mathbb{R}$;

$\rho_u, \rho_v, \rho_w \in [0, 2\pi)$;

where $\|\cdot\|$ denotes the Euclidean norm, ρ_u, ρ_v, ρ_w are the rotations around, and t_u, t_v, t_w are the translations along, the u-, v- and w-axis, respectively.

It is a nonlinear optimization problem. We use the `MATLAB` function *fminunc* to solve it. The solution is the best transformation corresponding to a given match $\mathbf{M}_k$.

C. Neighborhood Linkage Model

A neighborhood linkage model is used to combine the metrology data from different resolutions for predictions. Recall Reference [40] used a one-to-one linkage to link each low-accuracy output with a single high-accuracy output. In this metrology problem, however, measurement errors in both low-resolution and high-resolution datasets prevent us from finding the perfect match between low-resolution and high-resolution data points. When there is some small misalignment left after the matching and alignment procedure, we need a more robust linkage model than the one-to-one linkage. Therefore, we extend the linkage model to connect each high-resolution data point to all the low-resolution $\eta_l(\cdot)$ in its neighborhood, conditioned on a given match, as follows:

$$y_h(\mathbf{x}_i) = \alpha_1 \sum_{j=1,\ldots,m_l} K(\mathbf{x}_i, \mathbf{x}_j)\ \eta_l(\mathbf{x}_j) + \alpha_0 + e, \tag{4.7}$$

where $y_h(\mathbf{x}_i)$ is the univariate high-resolution response defined in the same way as the low-resolution one, i.e., $y_h(\mathbf{x}_i) = (\mathbf{a}_i^h)^T \mathbf{p}_j$; $\boldsymbol{\alpha} = (\alpha_1, \alpha_0)$ are the scale and location coefficients, respectively; $K(\cdot,\cdot)$ is a kernel function; e is the residual, assumed to be i.i.d. $N(0, \sigma_e^2)$.

We choose the tri-cube kernel function but generalize it by having different kernel

widths $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_d)$ for different axes. The resulting kernel function is as follows:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \Psi\left(\sum_{k=1}^{d}\left(\frac{x_{ki} - x_{kj}}{\lambda_k}\right)^2\right) \tag{4.8}$$

with

$$\Psi(t) = \begin{cases} (1 - t^{3/2})^3 & \text{if} \quad t \leq 1; \\ 0 & \text{if} \quad t > 1. \end{cases} \tag{4.9}$$

This kernel function smoothly defines a neighborhood of a high-resolution data point. Only the low-resolution data points in the neighborhood are linked with the high-resolution data point. The size of the neighborhood is controlled by $\boldsymbol{\lambda}$ and will be decided by the data.

Utilizing the linkage model, we can predict the high-resolution response $y_h(\mathbf{x}_0)$ at any given location $\mathbf{x}_0$. Given the filtered low-resolution responses $\boldsymbol{\eta}_l$ at m_l locations, and conditioned on the model parameters $\boldsymbol{\theta}_h = (\boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda})$ and the match $\mathbf{M}_k$, $y_h(\mathbf{x}_0)$ has the following distribution

$$(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda}) \sim N(\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x}_0)\boldsymbol{\alpha}, \sigma_e^2), \tag{4.10}$$

where $\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x}_0)$ is a row vector, defined as $\left(\sum_{i=1,\ldots,m_l} K(\mathbf{x}_0, \mathbf{x}_i)\ \eta_l(\mathbf{x}_i), 1\right)$ for all $l = 1, 2, \ldots, L$. Denote the locations of high-resolution responses by $\mathbf{X}_h = (\mathbf{x}_1, \ldots, \mathbf{x}_{m_h})^T$. The distribution in (4.10) can be extended to the multivariate case, where $\mathbf{y}_h = (y_h(\mathbf{x}_1), \ldots, y_h(\mathbf{x}_{m_h}))^T$ as follows:

$$(\mathbf{y}_h|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda}) \sim N(\mathbf{F}_{\boldsymbol{\lambda}}\boldsymbol{\alpha}, \sigma_e^2\mathbf{I}), \tag{4.11}$$

where $\mathbf{F}_{\boldsymbol{\lambda}}$ abbreviates $\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{X}_h)$, an $m_h \times 2$ matrix whose j-th row is defined as $\left(\sum_{i=1,\ldots,m_l} K(\mathbf{x}_j, \mathbf{x}_i)\ \eta_l(\mathbf{x}_i), 1\right)$, $\mathbf{x}_i$ is the i-th row of $\mathbf{X}_l$, and $\mathbf{x}_j$ is the j-th row of $\mathbf{X}_h$.

D. Bayesian Priors for the Predictive Model

This section discusses how we choose the priors for the parameters in the Bayesian hierarchical model. The parameters consist of two parts: $(\boldsymbol{\theta}_l, \boldsymbol{\theta}_h)$, where $\boldsymbol{\theta}_l = (\boldsymbol{\beta}_l, \sigma_l^2, \kappa_l^2, \boldsymbol{\nu}_l)$ are the parameters involved in the low-resolution data model, and $\boldsymbol{\theta}_h = (\boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda})$ are the parameters involved in the neighborhood linkage model. They can also be grouped into three categories: the mean-component parameters $(\boldsymbol{\beta}_l, \boldsymbol{\alpha})$, the variance parameters $(\sigma_l^2, \kappa_l^2, \sigma_e^2)$, as well as the parameters $(\boldsymbol{\nu}_l, \boldsymbol{\lambda})$ in the correlation function and the kernel function, respectively.

Regarding the GP model for the low-resolution data, we follow the common practice (e.g., in [40]) and use a normal distribution for the prior of the mean parameter $\boldsymbol{\beta}_l$, inverse-gamma distributions for the priors of the variance parameters (σ_l^2, κ_l^2), and gamma distributions for the priors of the correlation parameter $\boldsymbol{\nu}_l$.

The linkage model is essentially a linear regression model once the kernel width $\boldsymbol{\lambda}$ is given. We choose the noninformative priors for $\boldsymbol{\alpha}$ and σ_e^2 to reflect the limited knowledge regarding how the two-resolution metrology data are related. An accompanying benefit of using the noninformative priors is that the resulting posterior distributions of $\boldsymbol{\alpha}$ and σ_e^2 are in closed forms. It helps speed up the computation in the subsequent Bayesian inference. The prior distribution for the kernel width $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_d)$ is chosen to be uniform over a range. Let λ_i follow a discrete distribution taking values $j \times c$ for $j = 1, \ldots, \lambda_0$, where c is the size of the increment, and λ_0 is a positive, typically large, integer. Assuming $\boldsymbol{\lambda}$ has a discrete distribution helps simplify the computation in the subsequent Bayesian inference as well.

Furthermore, the joint prior distribution is assumed to be the product of the

prior distributions of individual parameters as follows:

$$p(\boldsymbol{\theta}) = p(\boldsymbol{\beta}_l)p(\sigma_l)p(\kappa_l)p(\boldsymbol{\alpha})p(\sigma_h)p(\boldsymbol{\nu}_l)p(\boldsymbol{\lambda}) \tag{4.12}$$

where

$$\boldsymbol{\beta}_l \sim N(\boldsymbol{\mu}_l, \mathbf{Q}_l);$$

$$\sigma_l^2 \sim \text{Inv-Gamma}(a_1, b_1);$$

$$\kappa_l^2 \sim \text{Inv-Gamma}(a_2, b_2);$$

$$\nu_i \sim \text{Gamma}(a_3, b_3) \qquad \text{for } i = 1, \ldots, d;$$

$$p(\boldsymbol{\alpha}) \propto 1;$$

$$p(\sigma_e^2) \propto \sigma_e^{-2};$$

$$p(\lambda_i = j \times c) = 1/\lambda_0 \qquad \text{for } j = 1, 2, \ldots, \lambda_0, \text{ and } i = 1, \ldots, d.$$

The covariance matrix $\mathbf{Q}_l$ is a diagonal matrix, whose i-th diagonal elements q_i^2 denotes the variance for the i-th element of $\boldsymbol{\beta}_l$.

Here we include some discussion on how to choose the parameters in the aforementioned prior distributions. The $\boldsymbol{\beta}_l$ includes the parameters accounting for a part's actual location (on a measuring platform) and dimension. In Fig. 9, the dashed-line geometry $\mathbf{f}(\cdot, \boldsymbol{\beta}_l)$ deviates slightly in location from the nominal geometry. The deviation is small due to the use of the soft-fixturing process. There is also some difference in dimensions between these two geometries, but it is small because a manufacturing process normally produces the required dimension with reasonable accuracy. By treating the location of the nominal geometry to be at the origin of the coordinate system, we can assign the mean component of the prior distribution of $\boldsymbol{\beta}_l$ as $\boldsymbol{\mu}_l = (\mathbf{0}, \boldsymbol{\varphi}^*)$ where $\boldsymbol{\varphi}^*$ is the nominal dimensions of the part. The variance q_i^2 can be determined from a crude least-squares estimation of the part's location as well as from the typical manufacturing process capability [[26], Page 85] which accounts for variability in actual part's dimensions.

For the prior distribution of σ_l^2 and κ_l^2, we choose $\alpha_1 = \alpha_2 = 1$ and $\beta_1 = \beta_2 = 1 \times 10^{-4}$ so that σ_l and κ_l mostly takes values smaller than .2 millimeter (i.e., 200 microns). This is consistent with engineers' knowledge on the error magnitudes from the manufacturing and the measuring processes. Also, this prior distribution gives higher probabilities to the smaller values as they correspond to a better fit of the low-resolution data to the model. The prior chosen for each correlation parameter ν_i is Gamma(.01, .01), a noninformative prior [22].

E. Bayesian Prediction and Model Averaging

The ultimate goal of combining the two-resolution data is to predict the response $y_h(\mathbf{x}_0)$ at any input location $\mathbf{x}_0$ based on the observed data $\mathbf{y}_h$ and $\mathbf{y}_l$.

It can be shown (please see Section IV.E.1 for derivation) that given the prior distribution as specified in (4.12), the following predictive distribution holds true:

$$p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_l, \mathbf{y}_h) = \int_{\boldsymbol{\theta}_l} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h) p(\boldsymbol{\eta}_l|\boldsymbol{\theta}_l, \mathbf{y}_l) p(\boldsymbol{\theta}_l|\mathbf{y}_l) d\boldsymbol{\theta}_l, \tag{4.13}$$

where $p(\boldsymbol{\eta}_l|\boldsymbol{\theta}_l, \mathbf{y}_l)$ is given in (4.4), $p(\boldsymbol{\theta}_l|\mathbf{y}_l) \propto p(\boldsymbol{\theta}_l) \cdot p(\mathbf{y}_l|\boldsymbol{\theta}_l)$, and

$$p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h) = \sum_{\substack{\lambda_1 = c, 2c, \ldots, \lambda_0 c \\ \vdots \\ \lambda_d = c, 2c, \ldots, \lambda_0 c}} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l, \boldsymbol{\lambda}) p(\boldsymbol{\lambda}|\mathbf{y}_h, \boldsymbol{\eta}_l), \tag{4.14}$$

in which

$$(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l, \boldsymbol{\lambda}) \sim t_{m_h - 2}(\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x}_0)\hat{\boldsymbol{\alpha}}, s^2(1 + \mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x_0})(\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}})^{-1}\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x_0})^T)), \tag{4.15}$$

$$p(\boldsymbol{\lambda}|\mathbf{y}_h, \boldsymbol{\eta}_l) \propto p(\boldsymbol{\lambda})|\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}}|^{-\frac{1}{2}} \left[\frac{(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})}{2}\right]^{-\frac{m_h}{2}+1}, \tag{4.16}$$

$\mathbf{F}_{\boldsymbol{\lambda}}$ and $\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x}_0)$ are defined in (4.10) and (4.11),

$$\hat{\boldsymbol{\alpha}} = (\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}})^{-1}\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{y}_h,$$

$$s^2 = \frac{1}{m_h - 2}(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}}).$$

We can use the folowing algorithm to approximate $p(y_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h)$:

(i). Generate $(\boldsymbol{\theta}_l^{(1)}, \boldsymbol{\theta}_l^{(2)}, \ldots, \boldsymbol{\theta}_l^{(n)})$ from the posterior distribution $p(\boldsymbol{\theta}_l|\mathbf{y}_l)$ using MCMC algorithms [22]. In particular, this dissertation uses Gibbs sampler and slice sampling. An introduction to MCMC as well as some implementation details for this dissertation are provided in Appendix B. Reference [1] also adopted slice sampling scheme for Gaussian process models.

(ii). Approximate $p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_l, \mathbf{y}_h)$ by

$$p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_l, \mathbf{y}_h) = \frac{1}{n}\sum_{i=1}^{n} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h)p(\boldsymbol{\eta}_l|\boldsymbol{\theta}_l^{(i)}, \mathbf{y}_l) \tag{4.17}$$

where $p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h)$ and $p(\boldsymbol{\eta}_l|\boldsymbol{\theta}_l^{(i)}, \mathbf{y}_l)$ are given in (4.14) and (4.4), respectively.

The matching and alignment method in Section IV.B produces K matches and their corresponding transformations. Here we denote both a match and its corresponding transformation with $\mathbf{M}_k, k = 1, 2, ..., K$ and use a Bayesian model averaging [41] to combine the final predictions over different matches. A brief introduction to Bayesian model averaging is provided in Appendix C. Using Bayesian model averaging takes into account the uncertainty in the match between two-resolution data, and assign different weights, i.e., posterior probabilities, to the matches according to how much they are consistent with observed data. Then the Bayesian model averaging averages the predictive model in (4.13) over the pool of candidate matches $\{\mathbf{M}_k\}_{k=1,...,K}$, weighted by the posterior probabilities $p(\mathbf{M}_k|\mathbf{y}_h, \mathbf{y}_l)$, $k = 1, ..., K$. We

obtain the posterior distribution of $y_h(\mathbf{x}_0)$ given the two-resolution data $\mathbf{y}_h$ and $\mathbf{y}_l$ as:

$$p(y_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h) = \sum_{k=1}^{K} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_l, \mathbf{y}_h) p(\mathbf{M}_k|\mathbf{y}_l, \mathbf{y}_h), \tag{4.18}$$

where

$$p(\mathbf{M}_k|\mathbf{y}_h, \mathbf{y}_l) = \frac{p(\mathbf{y}_h|\mathbf{M}_k, \mathbf{y}_l) p(\mathbf{M}_k)}{\sum_{k=1}^{K} p(\mathbf{y}_h|\mathbf{M}_k, \mathbf{y}_l) p(\mathbf{M}_k)}, \tag{4.19}$$

and

$$p(\mathbf{y}_h|\mathbf{M}_k, \mathbf{y}_l) = \int_{\boldsymbol{\theta}_l} p(\mathbf{y}_h|\mathbf{M}_k, \boldsymbol{\eta}_l) p(\boldsymbol{\eta}_l|\mathbf{y}_l, \boldsymbol{\theta}_l) d\boldsymbol{\theta}_l, \tag{4.20}$$

$$p(\mathbf{y}_h|\mathbf{M}_k, \boldsymbol{\eta}_l) = \sum_{\substack{\lambda_1=1,2,\ldots,\lambda_0 \\ \vdots \\ \lambda_d=1,2,\ldots,\lambda_0}} (2\pi)^{-\frac{m_h}{2}+1} |\mathbf{F}_{\boldsymbol{\lambda}}^T \mathbf{F}_{\boldsymbol{\lambda}}|^{-\frac{1}{2}} \Gamma\left(\frac{m_h}{2} - 1\right) \left[\frac{(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T (\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})}{2}\right]. \tag{4.21}$$

The derivation of (4.21) is included in Subsection IV.E.2.Fig. 25 summarizes how the Bayesian model averaging procedure works.

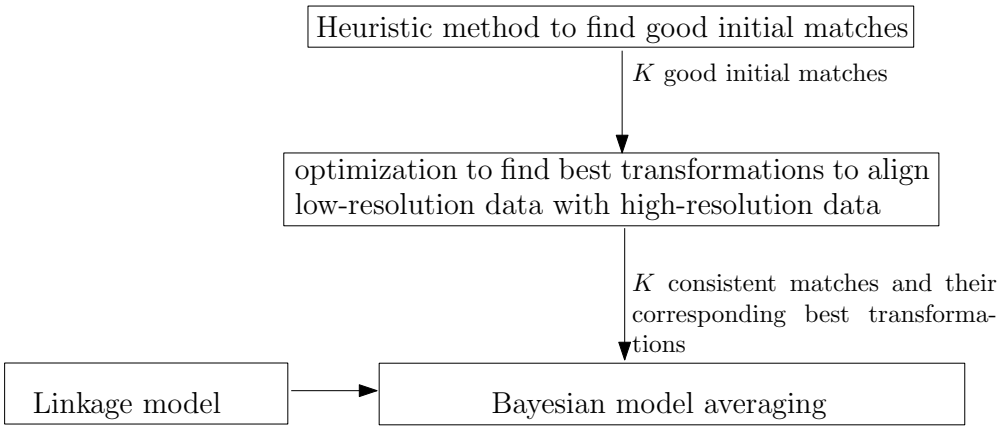

Fig. 25. Procedure for matching the two-resolution metrology data and the subsequent Bayesian model averaging for prediction

1. Proof of (4.13)

Recall the model parameters are $\boldsymbol{\theta} = (\boldsymbol{\theta}_l, \boldsymbol{\theta}_h)$. We express $p(y_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h)$ as

$$
\begin{aligned}
& p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_l, \mathbf{y}_h) \qquad (4.22)\\
&= \int_{\boldsymbol{\theta}_l} \int_{\boldsymbol{\theta}_h} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_l, \boldsymbol{\theta}_h) p(\boldsymbol{\theta}_l, \boldsymbol{\theta}_h|\mathbf{M}_k, \mathbf{y}_l, \mathbf{y}_h) d\boldsymbol{\theta}_h d\boldsymbol{\theta}_l \\
&= \int_{\boldsymbol{\theta}_l} \int_{\boldsymbol{\theta}_h} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l, \boldsymbol{\theta}_h) p(\boldsymbol{\theta}_h|\mathbf{M}_k, \boldsymbol{\eta}_l(\mathbf{X}_l), \mathbf{y}_h) d\boldsymbol{\theta}_h p(\boldsymbol{\eta}_l|\mathbf{y}_l, \boldsymbol{\theta}_l) p(\boldsymbol{\theta}_l|\mathbf{y}_l) d\boldsymbol{\theta}_l \\
&= \int_{\boldsymbol{\theta}_l} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l) p(\boldsymbol{\eta}_l|\mathbf{M}_k, \mathbf{y}_l, \boldsymbol{\theta}_l) p(\boldsymbol{\theta}_l|\mathbf{y}_l) d\boldsymbol{\theta}_l,
\end{aligned}
$$

where

$$
\begin{aligned}
& p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h) \qquad (4.23)\\
&\equiv \int_{\boldsymbol{\theta}_h} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l, \boldsymbol{\theta}_h)\ p(\boldsymbol{\theta}_h|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h) d\boldsymbol{\theta}_h \\
&= \int_{\boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda}} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda})\ p(\boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda}|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h)\ d\boldsymbol{\alpha}\ d\sigma_e^2\ d\boldsymbol{\lambda} \\
&= \int_{\boldsymbol{\lambda}} \int_{\sigma_e^2, \boldsymbol{\alpha}} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda})\ p(\boldsymbol{\alpha}, \sigma_e^2|\mathbf{M}_k, \boldsymbol{\lambda}, \mathbf{y}_h, \boldsymbol{\eta}_l)\ d\boldsymbol{\alpha}\ d\sigma_e^2\ p(\boldsymbol{\lambda}|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h) d\boldsymbol{\lambda}
\end{aligned}
$$

In order to derive $p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h)$, perform the integration in (4.23) in the following two steps:

(i) Integrate out $\boldsymbol{\alpha}$ and σ_e^2;

(ii) Integrate out $\boldsymbol{\lambda}$.

Step (i) integrate out $\boldsymbol{\alpha}$ and σ_e^2. We denote the inner integration in (4.23) by

$p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l, \boldsymbol{\lambda})$, that is,

$$\begin{aligned} & p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l, \boldsymbol{\lambda}) \\ & \equiv \int_{\sigma_e^2, \boldsymbol{\alpha}} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda})\ p(\boldsymbol{\alpha}, \sigma_e^2|\mathbf{M}_k, \boldsymbol{\lambda}, \mathbf{y}_h, \boldsymbol{\eta}_l)\ d\boldsymbol{\alpha}\ d\sigma_e^2 \\ & \propto \int_{\sigma_e^2, \boldsymbol{\alpha}} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda}) p(\mathbf{y}_h|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda})\ p(\boldsymbol{\alpha}, \sigma_e^2)\ d\boldsymbol{\alpha}\ d\sigma_e^2. \end{aligned} \tag{4.24}$$

Given the kernel width $\boldsymbol{\lambda}$, the linkage model can be considered as a linear regression model $\mathbf{y}_h = \mathbf{F}_{\boldsymbol{\lambda}}\boldsymbol{\alpha} + \boldsymbol{\epsilon}_h$. Recall $\boldsymbol{\epsilon}_h \sim N(\mathbf{0}, \sigma_e^2\mathbf{I})$. Therefore,

$$\begin{aligned} (\mathbf{y}_h|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda}) &\sim N(\mathbf{F}_{\boldsymbol{\lambda}}\boldsymbol{\alpha}, \sigma_e^2\mathbf{I}), \\ (y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda}) &\sim N(\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x}_0)\boldsymbol{\alpha}, \sigma_e^2). \end{aligned}$$

These are the same results as in (4.10) and (4.11). Reference [22] (Page 359) stated if the prior distribution of $\boldsymbol{\alpha}$ and σ_e^2 is $p(\boldsymbol{\alpha}, \sigma_e^2) \propto \sigma_h^{-2}$, the posterior predictive distribution of $y_h(\mathbf{x}_0)$, conditioned on the data and kernel width $\boldsymbol{\lambda}$, would be:

$$(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l(\mathbf{X}_0), \boldsymbol{\lambda}) \sim t_{m_h-2}(\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x}_0)\hat{\boldsymbol{\alpha}}, s^2(1 + \mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x_0})(\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}})^{-1}\mathbf{F}_{\boldsymbol{\lambda}}(\mathbf{x_0})^T)),$$

where $\hat{\boldsymbol{\alpha}} = (\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}})^{-1}\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{y}_h$ and $s^2 = \frac{1}{m_h-2}(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})$. This is how (4.15) is obtained. If we integrate out $\boldsymbol{\alpha}$ and σ_e^2, (4.23) becomes:

$$p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h) = \int_{\boldsymbol{\lambda}} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l(\mathbf{X}_0), \boldsymbol{\lambda})\ p(\boldsymbol{\lambda}|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h) d\boldsymbol{\lambda}. \tag{4.25}$$

Step (ii), integrate out $\boldsymbol{\lambda}$. Recall $\boldsymbol{\lambda}$ has a discrete distribution. Thus, the integration in (4.25) can be written as the following summation:

$$p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \boldsymbol{\eta}_l, \mathbf{y}_h) = \sum_{\substack{\lambda_1=1,2,\ldots,\lambda_0 \\ \vdots \\ \lambda_d=1,2,\ldots,\lambda_0}} p(y_h(\mathbf{x}_0)|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l, \boldsymbol{\lambda})\ p(\boldsymbol{\lambda}|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l),$$

where

$$p(\boldsymbol{\lambda}|\mathbf{M}_k, \mathbf{y}_h, \boldsymbol{\eta}_l) \propto p(\boldsymbol{\lambda})p(\mathbf{y}_h|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\lambda}). \tag{4.26}$$

This is how we obtain (4.14). The marginal distribution of the high-resolution data given the inputs $\boldsymbol{\eta}_l$ and the kernel width $\boldsymbol{\lambda}$ is as follows:

$$\begin{aligned}
&p(\mathbf{y}_h|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\lambda}) \qquad (4.27)\\
=&\int_{\sigma_e^2,\boldsymbol{\alpha}} p(\mathbf{y}_h, \boldsymbol{\alpha}, \sigma_e^2|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\lambda})\ d\boldsymbol{\alpha}\ d\sigma_e^2\\
=&\int_{\sigma_e^2,\boldsymbol{\alpha}} p(\mathbf{y}_h|\mathbf{M}_k, \boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda})\ p(\boldsymbol{\alpha}, \sigma_e^2)\ d\boldsymbol{\alpha}\ d\sigma_e^2\\
=&\int_{\sigma_e^2}\int_{\boldsymbol{\alpha}} (2\pi)^{-\frac{m_h}{2}}(\sigma_e^2)^{-\frac{m_h}{2}}\ exp\left\{-\frac{1}{2\sigma_e^2}(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\boldsymbol{\alpha})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\boldsymbol{\alpha})\right\}\ d\boldsymbol{\alpha}\ \sigma_h^{-2}d\sigma_e^2\\
=&\int_{\sigma_e^2}\int_{\boldsymbol{\alpha}} exp\left\{-\frac{1}{2\sigma_e^2}[(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}}) + (\boldsymbol{\alpha} - \hat{\boldsymbol{\alpha}})^T\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}}(\boldsymbol{\alpha} - \hat{\boldsymbol{\alpha}})]\right\}\ d\boldsymbol{\alpha}\\
&\cdot (2\pi)^{-\frac{m_h}{2}}(\sigma_e^2)^{-\frac{m_h}{2}-1}d\sigma_e^2\\
=&\int_{\sigma_e^2}\int_{\boldsymbol{\alpha}} exp\left\{-\frac{1}{2\sigma_e^2}(\boldsymbol{\alpha} - \hat{\boldsymbol{\alpha}})^T\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}}(\boldsymbol{\alpha} - \hat{\boldsymbol{\alpha}})\right\}\ d\boldsymbol{\alpha}\\
&\cdot (2\pi)^{-\frac{m_h}{2}} exp\left\{-\frac{1}{2\sigma_e^2}(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})\right\}(\sigma_e^2)^{-\frac{m_h}{2}-1}d\sigma_e^2\\
=&\int_{\sigma_e^2} 2\pi|\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}}|^{-\frac{1}{2}}\sigma_e^2(2\pi)^{-\frac{m_h}{2}} exp\left\{-\frac{1}{2\sigma_e^2}(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})\right\}(\sigma_e^2)^{-\frac{m_h}{2}-1}d\sigma_e^2\\
=&(2\pi)^{-\frac{m_h}{2}+1}|\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}}|^{-\frac{1}{2}}\int_{\sigma_e^2} (\sigma_e^2)^{-(\frac{m_h}{2}-1+1)} exp\left\{-\frac{(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})}{2\sigma_e^2}\right\}d\sigma_e^2\\
=&(2\pi)^{-\frac{m_h}{2}+1}|\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}}|^{-\frac{1}{2}}\Gamma\left(\frac{m_h}{2} - 1\right)\left[\frac{(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})}{2}\right]^{-\frac{m_h}{2}+1}.
\end{aligned}$$

Note that in the second line of the above derivation, we utilize (4.11) which specifies the distribution of $p(\mathbf{y}_h|\boldsymbol{\eta}_l, \boldsymbol{\alpha}, \sigma_e^2, \boldsymbol{\lambda})$. Given the above, (4.26) can now be written as

$$p(\boldsymbol{\lambda}|\mathbf{y}_h, \boldsymbol{\eta}_l) \propto p(\boldsymbol{\lambda})|\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}}|^{-\frac{1}{2}}\left[\frac{(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h - \mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})}{2}\right]^{-\frac{m_h}{2}+1}.$$

This finishes the proof for (4.13).

2. Proof of (4.21)

To prove (4.21), we just integrate (4.27) over λ. As λ takes discrete values, we end up with a summation over all the possible values for λ:

$$p(\mathbf{y}_h|\mathbf{M}_k,\boldsymbol{\eta}_l)=\sum_{\substack{\lambda_1=1,2,\ldots,\lambda_0\\ \vdots\\ \lambda_d=1,2,\ldots,\lambda_0}}(2\pi)^{-\frac{m_h}{2}+1}|\mathbf{F}_{\boldsymbol{\lambda}}^T\mathbf{F}_{\boldsymbol{\lambda}}|^{-\frac{1}{2}}\Gamma(\frac{m_h}{2}-1)\big[\frac{(\mathbf{y}_h-\mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})^T(\mathbf{y}_h-\mathbf{F}_{\boldsymbol{\lambda}}\hat{\boldsymbol{\alpha}})}{2}\big].$$

This finishes the proof for (4.21).

CHAPTER V

NUMERICAL RESULTS

This chapter is devoted to validating the proposed Bayesian hierarchial model for combining two-resolution metrology data. It demonstrates the improvements achieved by the proposed method over alternative methods. The alternative methods include methods using either high-resolution data or low-resolution data alone and a method that combines data from different resolutions but does not address the misalignment issue.

The validation and comparison are performed for two different types of features: roundness features and a sine-wave feature. A roundness feature is a challenging feature even though it is two-dimensional. First, its v-axis coordinate can not be expressed as an explicit function of its u-axis. Second, a perfect circle would be invariant under rotation, and therefore matching and aligning a close-to-perfect circle is very difficult. A sine-wave feature is chosen because it is three dimensional and also a free-form feature. So this chapter presents two examples: the first example simulates two datasets of different resolutions measuring three circular features, respectively; and the second example uses measurements of a milled sine-wave surface (as illustrated in Fig. 2) from a CMM and an OCMM. Throughout this chapter, the values of physical quantities are expressed in millimeters unless indicated otherwise.

A. Circular Features

1. Simulate circular features

Circular features manufactured by turning processes are simulated using the formula proposed in [13] as follows:

$$u = u_0 + (r + A_1 \sin(4\tau) + A_2 \cos(3\tau) + A_3 \sin(7\tau) + A_4 \cos(10\tau) + \xi) \cos\tau, \quad (5.1)$$

$$v = v_0 + (r + A_1 \sin(4\tau) + A_2 \cos(3\tau) + A_3 \sin(7\tau) + A_4 \cos(10\tau) + \xi) \sin\tau, \quad (5.2)$$

where $A_1 \sin(4\tau) + A_2 \cos(3\tau) + A_3 \sin(7\tau) + A_4 \cos(10\tau)$ represents the low-frequency manufacturing errors showing some systematic pattern, ξ represents the high-frequency random manufacturing errors, and the other notations follow those in Fig. 26.

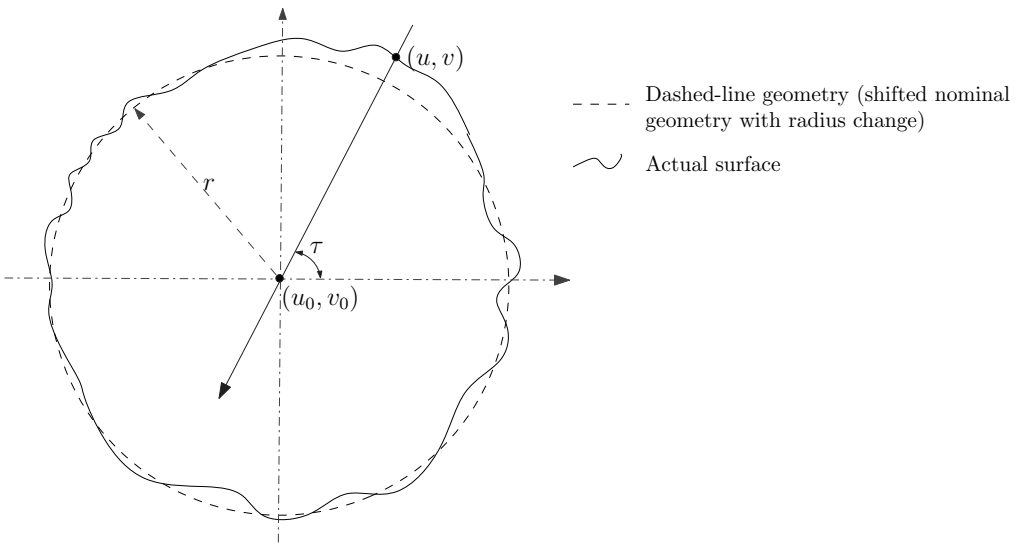

Fig. 26. Illustration of circle-related variables

In the simulation, (u_0, v_0) is set to be $(.05, .02)$, and the values are chosen to be within the typical range of the soft-fixturing errors. A_1, A_2, A_3 and A_4 are chosen

to be .03, −.02, −.01 and −.008, respectively, to reflect typical manufacturing errors of a turning process [[26], Page 85]. A standard deviation of .002 is chosen for ξ, according to the roughness level of a turning process [[26], Page 86]. In other words, ξ is simulated by $N(0, 4 \times 10^{-6})$.

Three circular features are simulated using (5.1) and (5.2) with the radius $r = 41$ and 100 and 150, respectively. Each of these simulated circles is treated as a true surface to be measured.

The high-resolution data are supposedly scattered over the surface, but each of them has little bias from the true surface. So m_h data points are first sampled from the simulated surface (via a Latin Hypercube sampling [47]), and then independent random noises ϵ_h with a variance of 1×10^{-6} (i.e., with a standard deviation of 1 micron) are added to them.

The low-resolution data are usually the averages of the surface coordinates in a spatial neighborhood within which a low-resolution device cannot distinguish distinct points. To simulate a device of resolution roughly 30 microns, a 30×30 microns moving window is first used to smooth the simulated surface. Then a dense sample of m_l data points is obtained, equally spaced over the whole surface. Finally, independent random noises with a variance of 9×10^{-6} (i.e., with a standard deviation of 3 microns) are added. Since low-resolution data may be misaligned from high-resolution data, a rigid body transformation is also performed to the low-resolution data. The simulation procedure is summarized in Fig. 27, and the data amounts of high- and low-resolution data are listed in Table VI.

2. Comparison results

The proposed Bayesian hierarchical model is used to predict the surface coordinates at locations where only low-resolution data are available. The resulting predictions

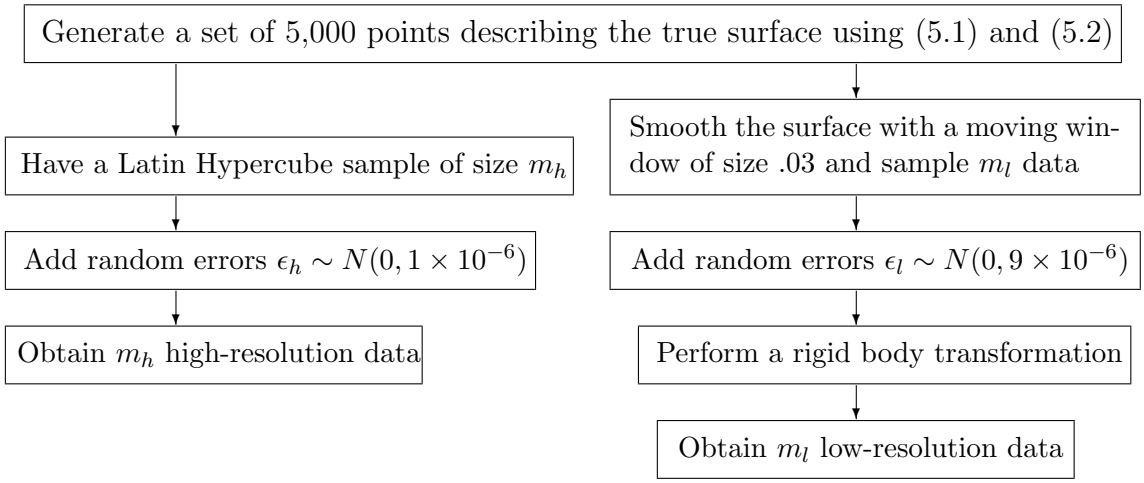

Fig. 27. Procedure to generate two-resolution data

are compared with three alternative approaches: the low-resolution data themselves, the predictions using the high-resolution data alone (based on the single-resolution model presented in Chapter III), and the predictions using the multi-resolution GP model [33], which does not have a matching and aligning procedure. For the last one, we use the `tgp` package [24] in `R`, which is close to the multi-resolution model presented in [33]. Since this is a simulation study, the true circular surface is known. The predicted surface from each method is compared with the true circular surface. Table VI presents the mean squared errors (MSE) of predictions, which are calculated by averaging the squared errors between the predicted and the true surfaces over different locations.

As one observes from Table VI, the proposed Bayesian hierarchical model (in the fifth column) outperforms all the alternative methods. For the three simulated circular features, the integrated prediction improves (in terms of the MSE values) over the observed low-resolution data by 74.31%, 70.86% and 52.86%, respectively. Another observation is that the predictions using the high-resolution data alone perform

Table VI. Evaluation and comparison of prediction accuracy for circular surfaces

radius	data amount (m_h/m_l)	MSE (mm^2)			
			prediction using data of		
		low-resolution	high-resolution	two-resolution with proposed method (having alignment)	two-resolution with multi-resolution GP (not having alignment)
41	20/512	3.83×10^{-5}	6.47×10^{-4}	9.84×10^{-6}	5.35×10^{-4}
100	20/1021	3.74×10^{-5}	1.11×10^{-3}	1.09×10^{-5}	1.98×10^{-3}
150	20/1013	3.84×10^{-5}	1.76×10^{-3}	1.81×10^{-5}	2.19×10^{-3}

worse than the low-resolution data. We believe this is because the high-resolution data are too scarce to make good predictions of the entire surface. The last column has the predictions from the `tgp` package which does not align data of different resolutions. When the two-resolution data are not aligned, the low-resolution data either contribute little to the combined prediction or mislead the integrated predictions. So the `tgp` package performs much worse than the proposed method. This result emphasizes the importance of addressing the misalignment problem when combining multi-resolution data for predictions.

B. Sine-wave Surface

1. Experiment setup

We have a manufactured part of size $101 \times 101 \times 51$, and its top surface is milled to be of a sine-wave shape. Fig. 28 shows the picture of the sine-wave part. The nominal geometry is $w = \psi_1 \sin(2\pi(\psi_2 + u)/\psi_3)$, where $\boldsymbol{\psi} = (\psi_1, \psi_2, \psi_3)$ are the dimension parameters, and ψ_1, ψ_2, and ψ_3 are called amplitude, phase, and wavelength,

respectively.

Two metrology devices are used to measure this sine-wave surface: the CMM is a Sheffield Discovery II D-8 with a TB 20 touch probe (Fig. 29), and the OCMM is a LDI Surveyor DS-2020 with a RPS 150 laser unit (Fig. 30). The CMM has a resolution of roughly 5 microns, while the OCMM has a resolution of about 50 microns. Even though this CMM does not have a resolution as high as many used in precision engineering, it serves as the high-resolution device in this particular pair. Moreover, the CMM and the OCMM have their own measuring platforms so that the part is repositioned to be measured on the second machine.

When using the OCMM, it results in a low-resolution dataset of $m_l = 1,560$ ($=40 \times 39$) points, which are evenly spaced over the surface with approximately a 2.54 mm inter-point distance in both u and v axes. The CMM measures the same surface and obtains a high-resolution dataset of also 1,560 points. The 1,560 high-resolution data are reserved as the representation of the "true" surface and will be used to as a benchmark to assess the prediction quality.

Here we consider seven different cases where the high-resolution observations are of different sizes ranging from $m_h = 10$ to $m_h = 80$. These observations are chosen via Latin hypercube sampling [47] from the 1,560 high-resolution data. For each case, the proposed Bayesian hierarchical model is used to make predictions combining the m_h high-resolution observations and the $m_l = 1,560$ low-resolution observations.

The position and orientation of the part on a metrology measuring platform are unknown. It may be different from its nominal position by the translation $\mathbf{t}$ and the rotation $\mathbf{H}$. For this reason, the coordinates of a point on the dashed-line geometry $\mathbf{f}(\cdot, \boldsymbol{\beta}_l)$ is $\mathbf{H}(u_i, v_i, w_i)^T + \mathbf{t}$, where (u_i, v_i, w_i) is a point on the nominal geometry satisfying the constraint $w_i = \psi_1 \sin(2\pi(\psi_2 + u_i)/\psi_3)$.

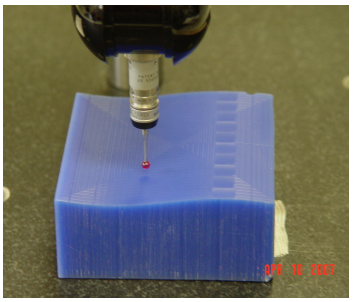

Fig. 28. Sine-wave part

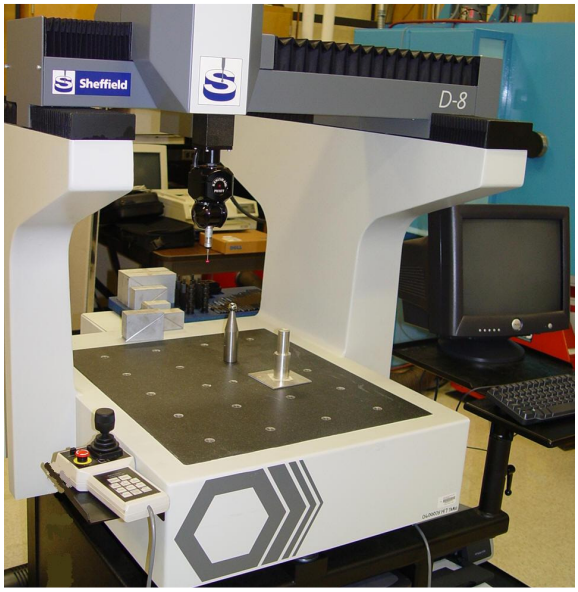

Fig. 29. CMM: Sheffield Discovery II D-8 with a TB 20 touch probe

Fig. 30. OCMM: LDI Surveyor DS-2020 with a RPS 150 laser unit

2. Predictive distributions

As in the simulated example, the proposed Bayesian hierarchical model is used to predict the part's surface at locations where only the low-resolution data are available. The predictions using the proposed method are compared with the observed low-resolution data, the predictions using the m_h high-resolution data alone, and the predictions using the `tgp` package which does not align the two-resolution data.

Fig. 31 shows the prediction results for one location on the sine-wave part and illustrates the benefit of combining two-resolution information. In Fig. 31, the "true" value (the black solid line) is actually a reserved high-resolution measurement. The grey solid curve represents the combined predictive distribution using the proposed multi-resolution method, and the grey line in the middle is the corresponding predicted median. The dashed-dotted line represents the observed low-resolution data. For this particular point, the combined prediction almost coincides with the "true" value and is much less biased than the low-resolution data. The dashed line and the dashed curve denote the predicted median and the predictive distribution using the m_h high-resolution observations, respectively. Using the two-resolution data also results in a much narrower distribution (which indicates a smaller uncertainty) than the distribution from using the high-resolution data alone. In summary, the combined prediction produces the best prediction, less biased (in terms of the distance between the predictive median and the true value) and with reduced uncertainty.

If the two-resolution datasets are combined to make predictions without properly aligning the datasets, it may not be able to produce the desired benefit. Still for the same location as in Fig. 31, Fig. 32 compares the predictive distribution from the proposed method with the predictive distribution from the `tgp` package, which does not align the two-resolution data. The proposed method predicts significantly better

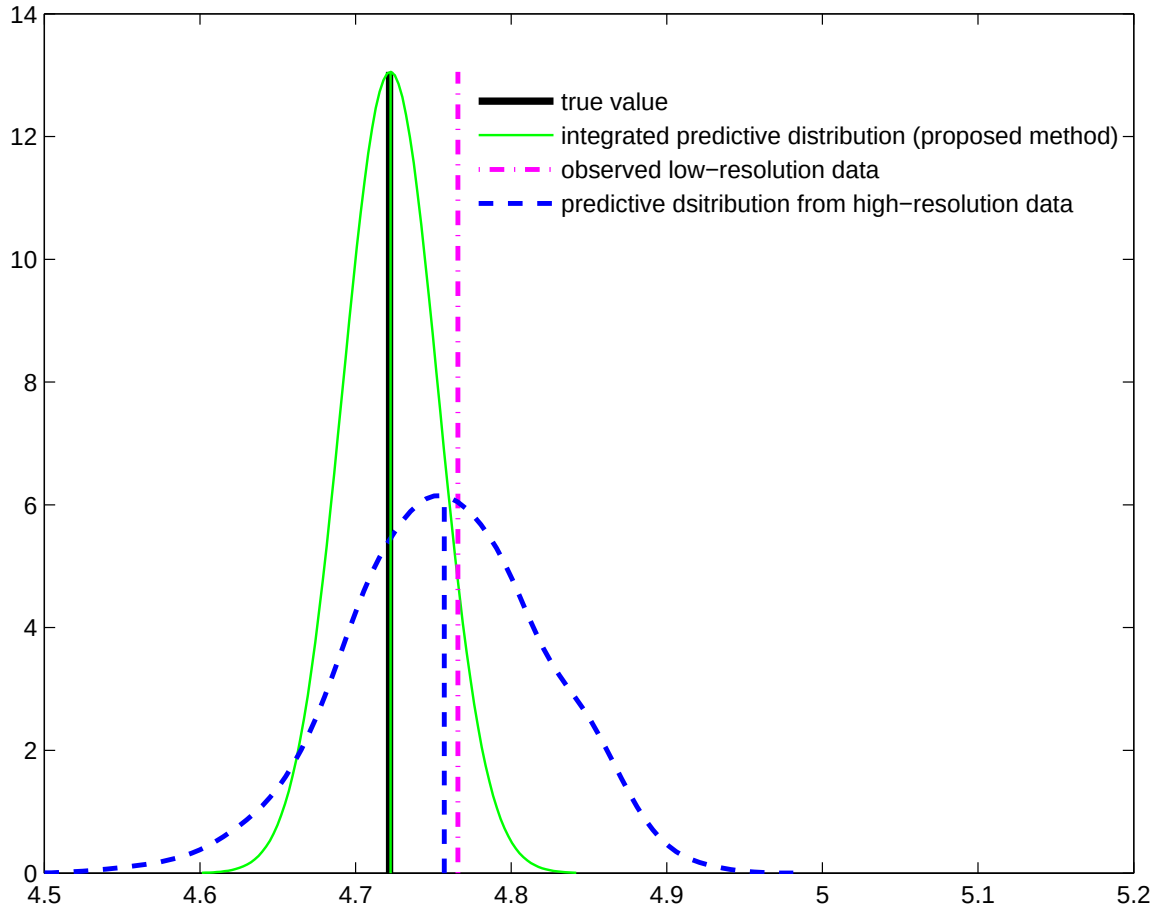

Fig. 31. Integrated prediction reduces bias and uncertainty

than the `tgp` package does.

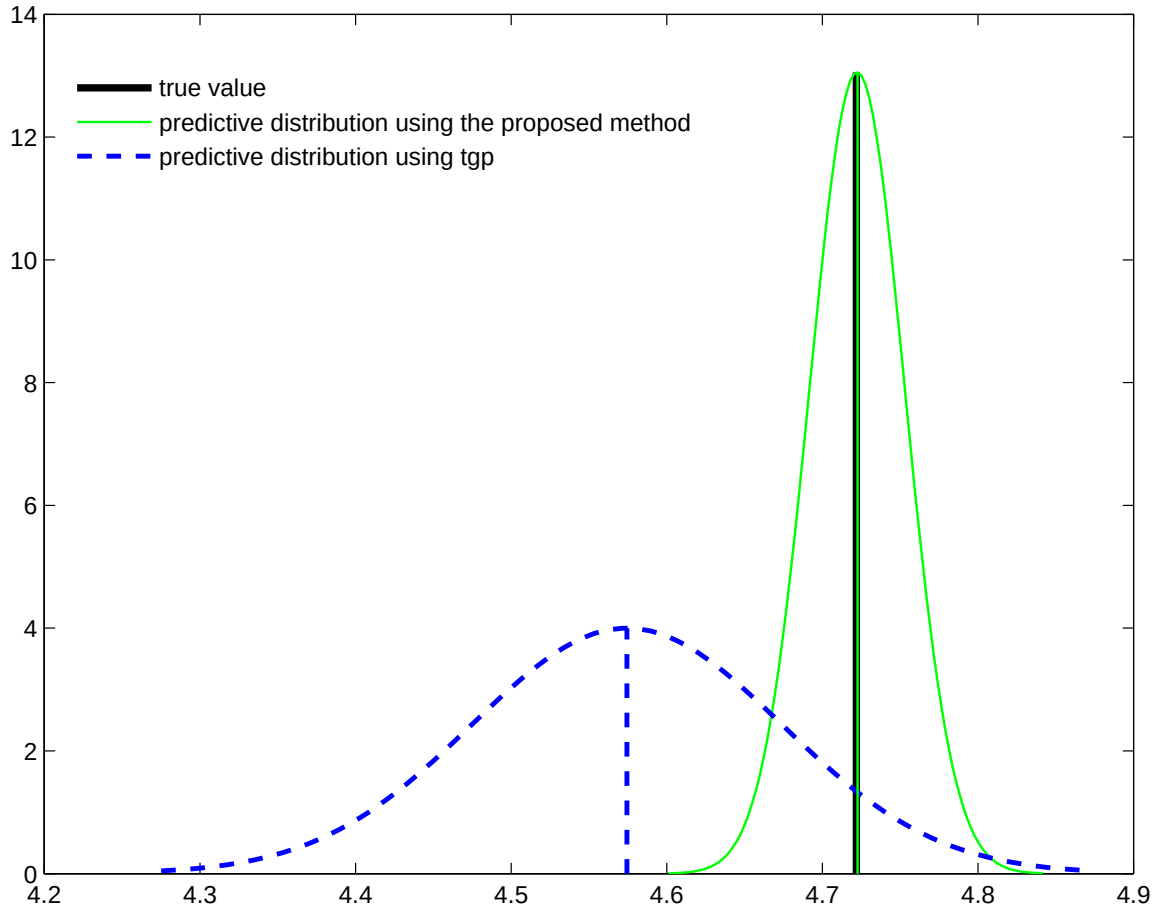

Fig. 32. Integrated predictions without addressing misalignment could lead to worse results

3. Comparison results

We use all of the aforementioned four methods to make predictions over the entire product surface and compare their predictions with the reserved high-resolution data, which serve as the "true" surface. Table VII summarizes the MSE values of the predictions for seven different cases. The cases differ by the amount of the observed high-resolution data. For example, "10/1560" refers to the case where 10

high-resolution data points and all the low-resolution data points, 1560 of them, are available.

Table VII. Evaluation and comparison of the predictions for the sine-wave surface

data amount (m_h/m_l)	MSE (mm^2)			
		prediction using data of		
	low-resolution	high-resolution	two-resolution with `tgp` package (not having alignment)	two-resolution with proposed method (having alignment)
10/1560	1.31×10^{-3}	2.08×10^{-2}	9.73×10^{-2}	1.03×10^{-3}
20/1560	1.18×10^{-3}	1.04×10^{-2}	8.79×10^{-2}	7.85×10^{-4}
30/1560	1.21×10^{-3}	3.94×10^{-3}	7.11×10^{-2}	7.52×10^{-4}
40/1560	1.05×10^{-3}	2.44×10^{-3}	6.10×10^{-2}	7.26×10^{-4}
50/1560	1.05×10^{-3}	2.11×10^{-3}	5.48×10^{-2}	6.74×10^{-4}
60/1560	1.05×10^{-3}	1.46×10^{-3}	4.36×10^{-2}	6.78×10^{-4}
80/1560	1.05×10^{-3}	1.45×10^{-3}	2.69×10^{-2}	7.04×10^{-4}

Supposedly, the difference between the "true" surface and the low-resolution data should be the same across the different cases. However, we observe that the MSE values listed in Table VII, Column 2 are different across the cases. This is because this sine-wave example is a real case study, and we do not know the true transformation between the low-resolution data and the high-resolution data. We have to estimate the transformation using the observed high-resolution and low-resolution data. As such, the estimated transformations differ when using different sets of high-resolution data. It leads to different MSE values for the low-resolution data, as shown in Table VII. To be fair, we consider the smallest MSE values that we find for the low-resolution data, 1.05×10^{-3}, as the true MSE value between the low-resolution data and the "true" surface.

Among the three alternative methods (Table VII, Column $2-4$), Column 2 has the smallest MSE values. It implies that the observed low-resolution data follow the "true" surface most closely.

Comparing the MSE values of the proposed method (Table VII, Column 5) with the smallest MSE value of the observed low-resolution data, 1.05×10^{-3}, we calculate the differences, divide them by 1.05×10^{-3}, and summarize the percentages in Table VIII, Column 2. Note 1.05×10^{-3} is regarded as the "true" MSE value of the observed low-resolution data. Other columns in Table VIII are calculated likewise.

Table VIII. Improvement in percentage of the proposed methods over other methods

Improvement over alternative methods			
data amount (m_h/m_l)	observed low-resolution data (1.05×10^{-3})	prediction using high-resolution data with GP model	prediction using two-resolution data with `tgp` package (not having alignment)
10/1560	1.90%	95.04%	98.94%
20/1560	25.24%	99.92%	99.11%
30/1560	28.38%	80.91%	98.94%
40/1560	30.86%	70.25%	98.81%
50/1560	35.80%	68.06%	98.77%
60/1560	35.42%	53.56%	98.44%
80/1560	32.95%	51.45%	97.38%

From Table VIII, Column 2, we notice that the proposed method produces MSE values at least 25% less than the observed low-resolution dataset does, when the high-resolution data are over 20. As the high-resolution data amount increases, the proposed method produces more reduced MSE values. However, the reduction does not grow in proportion to the increase in the high-resolution data amount. When

the high-resolution data are of size 50 or 60 (Table VIII, Column 2, Box $5-6$), the reduction reaches the maximum, about 35%.

In the case of "10/1560", the calculated reduction is only around 2% (Table VIII, Column 2, Box 1). Ten data points are not many enough to estimate the six parameters for rigid body transformation well, and therefore the low-resolution dataset is poorly aligned with the high-resolution dataset. Since we assume that the "true" surface is perfectly aligned with the high-resolution dataset, the low-resolution dataset is also poorly aligned with the "true" surface. As a result, the observed low-resolution dataset has a relatively large MSE value, 1.31×10^{-3}, after transformation (Table VII, Column 2, Box 1). This MSE value is significantly larger than 1.05×10^{-3}, the MSE value of the low-resolution dataset when we have more than 40 high-resolution data to estimate the transformation(Table VII, Column 2, Box $4-7$). If we calculate how much the proposed method reduces the MSE values of 1.31×10^{-3}, the reduction percentage would be 21.37%, for the case of "10/1560". This number is comparable with other percentages in Table VIII, Column 2.

The MSE values of using high-resolution data alone (Table VII, Column 3) are significantly larger than the MSE values of using two-resolution data with the proposed method (Table VII, Column 4). We take the differences between Column 3 and 4, divide them by Column 3 and record them in the third column of Table VIII. The calculated results show that the proposed method produces MSE values more than 51% less than the method of using high-resolution data alone does. As the amount of high-resolution data increases, the reduction percentage decreases because the method of using high-resolution data alone performs better when there are more high-resolution data available. This outcome makes sense because if we have plenty of high-resolution data, we will not need low-resolution data at all.

If we combine two-resolution data using the `tgp` package which does not address

the misalignment, the resulting MSE values (Table VII, Column 4) are normally two magnitudes bigger than the MSE values of the proposed method (Table VII, Column 5), which addresses the misalignment. In other words, the proposed method produces MSE values $97\% - 99\%$ smaller than those of the `tgp` package, as shown in the last column of Table VIII. Even when the high-resolution data amount reaches a relatively big value of 80, the above conclusion still holds true. This highlights the importance of addressing the misalignment problem. Otherwise, low-resolution data will either contribute little to the combined predictions or mislead them.

In summary, these results confirm the conclusions in the simulated examples. The proposed method performs significantly better than the methods using single-resolution data only as well as the method that combines two-resolution data but does not address the misalignment issue.

4. Estimation of kernel width

Recall that a discrete uniform distribution is assigned as the prior distribution for the kernel width $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)$, i.e., $p(\lambda_i = j \times c) = 1/\lambda_0$ for $j = 1, 2, \ldots, \lambda_0$ and $i = 1, 2$. In this example, we choose $\lambda_0 = 30$, and $c = 2.54$ which is the average inter-point distance in the low-resolution data. Fig. 33 shows the marginal posterior distributions of the kernel widths in the u- and the v-axis. The kernel width along the u-axis λ_1 is 2.54, suggesting that only the nearest low-resolution data point in the u-direction has a strong connection with each high-resolution data point. For the v-axis, the kernel width λ_2 of 2.54 has the highest probability, but the probabilities at the first eight kernel width have a noticeable nonzero mass. This makes good sense for the sine-wave surface in Fig. 2. It is easier to align the sine-wave surface along the u-direction than along the v-direction because of a much larger surface change in the u-axis than in the v-axis.

The Bayesian hierarchical method assigns posterior weights to different kernel widths according to how much they are consistent with the data. As shown in Fig. 33, the kernel widths that are greater than 20.32 (or eight units of inter-point distance) barely have any posterior weights and will not contribute to the posterior inference in the sine-wave example. This is not a surprise because the remaining misalignment is supposed to be small after all the alignment efforts. This implies that as long as we choose a big upper bound λ_0, the posterior inference of the prediction will not be sensitive to the prior choice for $\boldsymbol{\lambda}$.

We also conduct a sensitivity analysis to see how different increments c affect the combined predictions. The results for the case using 20 high-resolution data and 1560 low-resolution data are summarized in Table IX. It shows that the combined predictions are insensitive to the value of c when c takes a value less than or equal to the average distance, 2.54 mm, between low-resolution data points.

Table IX. Sensitivity analysis of the choices of c

	$c = i \times 2.54$ mm				
	$i = 1$	$i = 0.75$	$i = 0.5$	$i = 0.25$	$i = 0.1$
MSE	7.853×10^{-4}	7.782×10^{-4}	7.772×10^{-4}	7.818×10^{-4}	7.822×10^{-4}

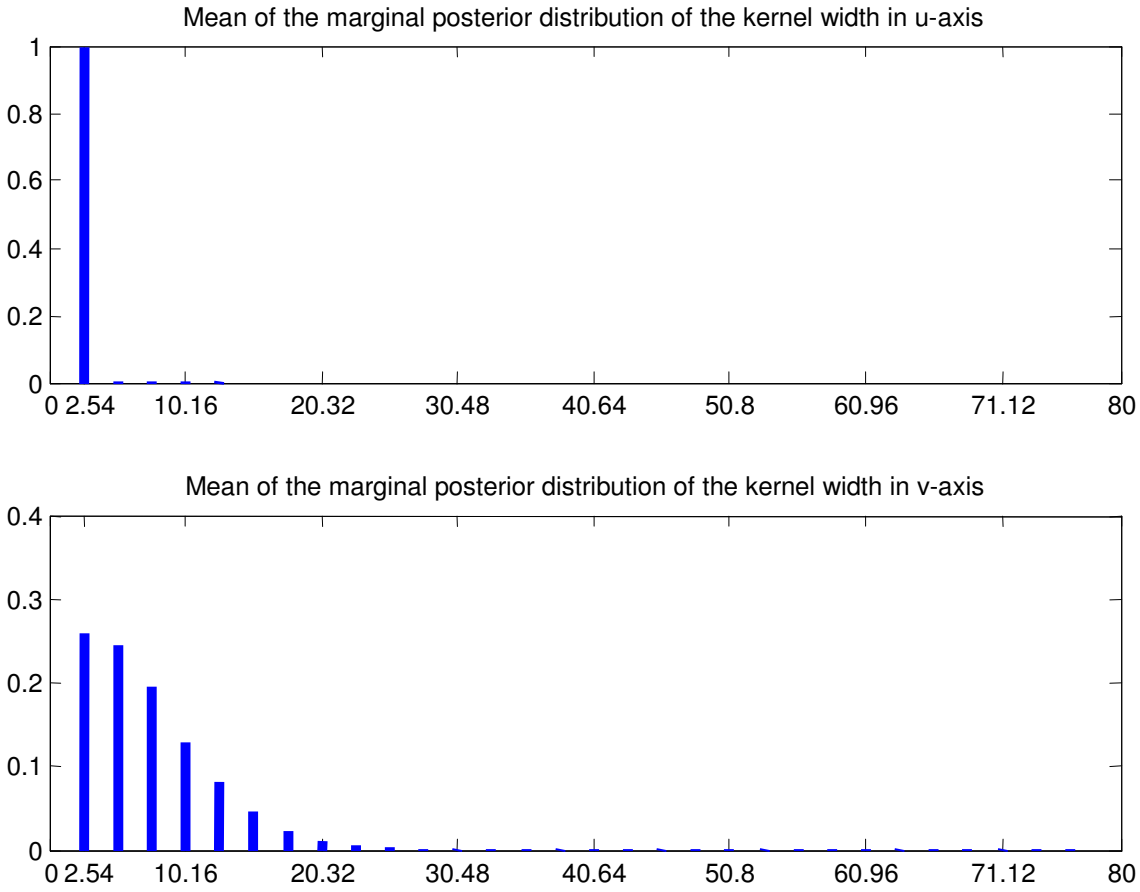

Fig. 33. Marginal posterior distributions of the kernel widthes in u-axis and v-axis

CHAPTER VI

SUMMARY AND FUTURE WORK

This chapter summarizes the contributions and findings of the dissertation. It also discusses the open issues that are worthy of further exploration.

A. Summary

This dissertation presents a Bayesian hierarchical model for combining two-resolution metrology data for the purpose of predicting a product surface. The proposed method also addresses the misalignment between two-resolution datasets.

The following summarizes the major contributions of the dissertation:

1. Gaussian process model for single-resolution metrology data

This dissertation develops a Gaussian process model for single-resolution metrology data. The model can work by itself for either high-resolution data or low-resolution data alone. It provides a better fit for metrology data by taking into account systematic patterns in a product surface caused by manufacturing errors. The model could also produce a predictive distribution of a product surface and therefore offer a probabilistic assessment of form error. The dissertation validates the single-resolution model using real metrology data and demonstrates its improvements over traditional methods, i.e., the MZ method and the OLS method, in terms of better estimating form errors. This single-resolution model serves as the low-resolution data model in the context of combining two-resolution data.

2. Bayesian hierarchical model for combining two-resolution metrology data

The dissertation also develops a Bayesian hierarchical model for combining two-resolution metrology data for the purpose of predicting a product surface. The developed method also takes the necessary steps to address the misalignment between the two-resolution datasets. A heuristic matching algorithm together with an optimization procedure finding the best transformation is used to align the data of different resolutions, and a neighborhood linkage model is established to handle the remaining mismatch after the alignment. The heuristic matching algorithm finds a set of consistent matches, and the Bayesian model averaging combines the final predictions over the consistent matches.

This dissertation also shows that the proposed method performs significantly better in terms of predicting accuracy and uncertainty than the methods that use only a single-resolution dataset and than the integrated method that does not align the two-resolution data.

B. Future Work

Based on the study in this dissertation, we find the following direction worthy of further exploration:

1. Effectiveness of combining two-resolution data

Although the proposed method can effectively combine data from two different resolutions, additional research questions remain: What factors affect the performance of the integrated analysis? What is the best strategy to achieve the highest performance? For example, the sine-wave example shows that when the amount of high-resolution data is doubled (i.e., the cost is almost doubled), the reduction in the MSE of the pre-

diction is far from being doubled. We believe that two factors are critical: the ratio of data between high- and low-resolution datasets as well as the resolution difference between them. A comprehensive research on this issue will provide insights for devising a strategy for when and how to collect more precious high-resolution data. Therefore, studying how to produce the most cost-effective benefit in combining multi-resolution data for prediction definitely merits further exploration.

2. Exploration of different modeling approaches and extension to multi-resolution levels

The current method uses a calibrating idea, as shown in Fig 34 (a). Looking at this two-resolution data problem from a slightly different aspect, it seems reasonable that metrology data from different resolutions are different reflections or representations of the same underlying true surface, illustrated in Fig. 34 (b). Following this idea, it would make sense to connect data of different resolutions through the underlying true surface that they measure. To be specific, the alternative approach could model the underlying physical surface with a spline, and establish the observations at different resolutions as the same physical surface being smoothed using kernel functions of different kernel widths. Then we can pools all the data together to estimate the unknown parameters involved in the spline model. One benefit of this alternative approach is that it would be relatively easier to extend the alternative approach to a multi-resolution model for combining data from more than two resolutions.

3. Other engineering applications

Many potential applications exist in remote sensing. For example, forestry scientists have to combine a scanning LIDAR (Light Detection And Ranging) with a profiling LIDAR to survey forestry better. A scanning LIDAR provides high-resolution data

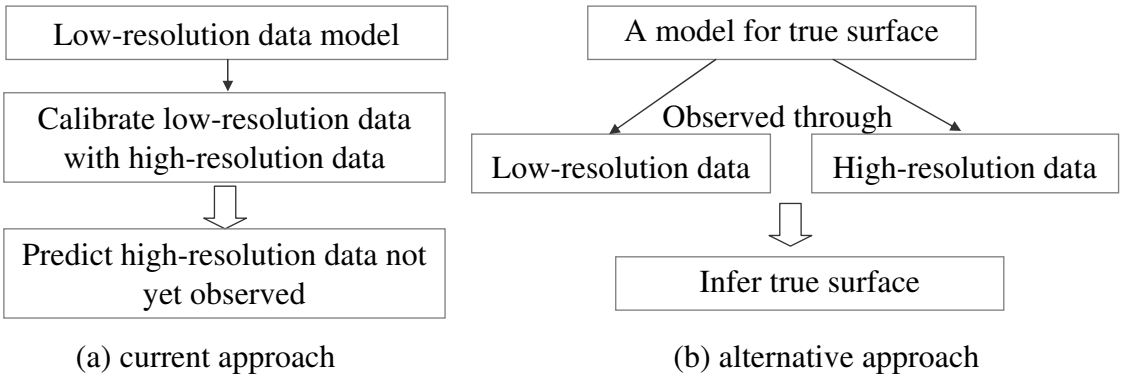

Fig. 34. An alternative idea for combining two-resolution data

covering a relatively small area of one square mile, while a profiling LIDAR gives low-resolution data covering a much bigger geographical area that could be as large as a county in Texas. New applications such as this will have their own needs, calling for new methods or novel usages of existing methods to address them.

REFERENCES

[1] S. Banerjee, B. Carlin, and A. Gelfand, *Hierarchical Modeling and Analysis for Spatial Data*. Boca Raton, Florida: Chapman & Hall/CRC, 2004.

[2] M. Bernardo, R. Buck, L. Liu, W. Nazaret, J. Sacks, and W. Welch, "Integrated circuit design optimization using a sequential strategy," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 11, no. 3, pp. 361–372, 1992.

[3] L. Brown, "A survey of image registration techniques," *ACM Computing Surveys*, vol. 24, pp. 326–376, 1992.

[4] V. Carbone, M. Carocci, E. Savio, G. Sansoni, and L. DE Chiffre, "Combination of a vision system and a coordinate measuring machine for the reverse engineering of freeform surfaces," *International Journal of Advanced Manufacturing Technology*, vol. 17, no. 4, pp. 263–271, 2001.

[5] S. Chang and S. Ho, "Multivariate statistical process control for inspection data from coordinate measuring machines," *International Journal of Industrial Engineering Theory, Applications and Practices*, vol. 8, pp. 347–358, 2001.

[6] C. Chen and C. Liu, "A study on analyzing the problem of the spherical form error," *Precision Engineering*, vol. 24, no. 2, pp. 119–126, 2000.

[7] L. Chen and G. Lin, "A vision-aided reverse engineering approach to reconstructing free-form surfaces," *Robotics and Computer-Integrated Manufacturing*, vol. 13, no. 4, pp. 323–336, 1997.

[8] N. Cho and J. Tu, "Roundness modeling of machined parts for tolerance analysis," *Precision Engineering*, vol. 25, no. 1, pp. 35–47, 2001.

[9] W. Choi and T. Kurfess, "Dimensional measurement data analysis, part 1: A zone fitting algorithm," *Journal of Manufacturing Science and Engineering*, vol. 121, pp. 238–245, 1999.

[10] ——, "Dimensional measurement data analysis, part 2: Minimum zone evaluation," *Journal of Manufacturing Science and Engineering*, vol. 121, pp. 246–250, 1999.

[11] N. Cressie and S. Lahiri, "Asymptotics for REML estimation of spatial covariance parameters," *Journal of Statistical Planning and Inference*, vol. 50, no. 3, pp. 327–341, 1996.

[12] G. Deng, G. Wang, and J. Duan, "A new algorithm for evaluating form error: the valid characteristic point method with the rapidly contracted constraint zone," *Journal of Materials Processing Tech.*, vol. 139, no. 1-3, pp. 247–252, 2003.

[13] M. Desta, H. Feng, and D. OuYang, "Characterization of general systematic form errors for circular features," *International Journal of Machine Tools and Manufacture*, vol. 43, no. 11, pp. 1069–1078, 2003.

[14] M. Dowling, P. Griffin, K. Tsui, and C. Zhou, "A comparison of the orthogonal least squares and minimum enclosing zone methods for form error estimation," *Manufacturing Review*, vol. 8, no. 2, pp. 120–138, 1995.

[15] ——, "Statistical issues in geometric feature inspection using coordinate measuring machines," *Technometrics*, vol. 39, no. 1, pp. 3–17, 1997.

[16] M. Dowling, P. Griffin, and C. Zhou, "Statistical issues in geometric tolerance verification using coordinate measuring machines," Georgia Institute of Technology, School of Industrial and Systems Engineering, Altanta, GA, Tech. Rep.,

1993.

[17] I. Dryden and K. Mardia, *Statistical Shape Analysis*. New York: John Wiley & Sons, 1998.

[18] K. Fan and J. Lee, "Analysis of minimum zone sphericity error using minimum potential energy theory," *Precision Engineering*, vol. 23, no. 1, pp. 65–72, 1999.

[19] J. Feng, H. Ip, L. Lai, and A. Linney, "Robust point correspondence matching and similarity measuring for 3D models by relative angle-context distributions," *Image and Vision Computing*, vol. 26, no. 6, pp. 761–775, 2008.

[20] H. D. Ferreira, M. and H. Lee, "Multi-scale random field models," Duke University, Statistics Department, Tech. Rep., 2005.

[21] M. Gaudard, M. Karson, E. Linder, and D. Sinha, "Bayesian spatial prediction," *Environmental and Ecological Statistics*, vol. 6, no. 2, pp. 147–171, 1999.

[22] A. Gelman, J. B. Carlin, H. S. Stein, and D. B. Rubin, *Bayesian Data Analysis*. Boca Raton, Florida: Chapman & Hall/CRC, 2003.

[23] C. Gotway and L. Young, "Combining incompatible spatial data." *Journal of the American Statistical Association*, vol. 97, no. 458, pp. 632–649, 2002.

[24] R. Gramacy and M. Taddy, *The tgp package*, 2008, [Online; accessed August 6, 2008]. Available: http://www.ams.ucsc.edu/∼rbgrammacy/tgp.html

[25] P. Green and K. Mardia, "Bayesian alignment using hierarchical models, with applications in protein bioinformatics," *Biometrika*, vol. 93, no. 2, pp. 235–254, 2006.

[26] M. Groover, *Fundamentals of Modern Manufacturing: Materials, Processes, and Systems*, 2nd ed. New York: John Wiley & Sons, 2004.

[27] R. Henke, K. Summerhays, J. Baldwin, R. Cassou, and C. Brown, "Methods for evaluation of systematic geometric deviations in machined parts and their relationships to process variables," *Precision Engineering*, vol. 23, no. 4, pp. 273–292, 1999.

[28] D. Higdon, M. Kennedy, J. Cavendish, J. Cafeo, and R. Ryne, "Combining field data and computer simulations for calibration," *SIAM Journal on Scientific Computing*, vol. 26, no. 2, pp. 448–466, 2004.

[29] J. Huang, "An exact minimum zone solution for sphericity evaluation," *Computer-Aided Design*, vol. 31, no. 13, pp. 845–853, 1999.

[30] ——, "An exact solution for the roundness evaluation problems," *Precision Engineering*, vol. 23, no. 1, pp. 2–8, 1999.

[31] F. Hulting, "Comment: An industrial view of coordinate measurement data analysis," *Statistica Sinica*, vol. 5, pp. 191–204, 1995.

[32] ——, "Discussion: Statistical issues in geometric feature inspection using coordinate measuring machines," *Technometrics*, vol. 39, no. 1, pp. 18–20, 1997.

[33] M. Kennedy and A. O'Hagan, "Predicting the output from a complex computer code when fast approximations are available," *Biometrika*, vol. 87, no. 1, pp. 1–13, 2000.

[34] ——, "Bayesian calibration of computer models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 3, pp. 425–464, 2001.

[35] T. Kurfess, D. Banks, and L. Wolfson, "A multivariate statistical approach to metrology," *Journal of Manufacturing Science and Engineering*, vol. 118, p. 652, 1996.

[36] R. Mitra and N. Murthy, "Elastic, maximal matching," *Pattern Recognition*, vol. 24, no. 8, pp. 747–753, 1991.

[37] S. Motavalli, V. Suharitdamrong, and A. Alrashdan, "Design model generation for reverse engineering using multi-sensors," *IIE Transactions*, vol. 30, no. 4, pp. 357–366, 1998.

[38] V. Namboothiri and M. Shunmugam, "Form error evaluation using L1-approximation," *Computer Methods in Applied Mechanics and Engineering*, vol. 162, no. 1-4, pp. 133–149, 1998.

[39] R. Neal, "Slice sampling," *Annals of Statistics*, vol. 31, no. 3, pp. 705–767, 2003.

[40] Z. Qian and C. Wu, "Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments," *Technometrics*, vol. 50, pp. 192–205, 2008.

[41] A. Raftery, D. Madigan, and J. Hoeting, "Bayesian model averaging for linear regression models," *Journal of the American Statistical Association*, vol. 92, no. 437, pp. 179–191, 1997.

[42] S. Ranade and A. Rosenfeld, "Point pattern matching by relaxation," *Pattern Recognition*, vol. 12, pp. 269–275, 1980.

[43] C. Reese, A. Wilson, M. Hamada, H. Martz, and K. Ryan, "Integrated analysis of computer and physical experiments," *Technometrics*, vol. 46, no. 2, pp. 153–164, 2004.

[44] Y. Ren, "Addaptive evolutionary monte carlo for optimization," Ph.D. dissertation, Texas A&M University, College Station, TX, 2008.

[45] G. Samuel and M. Shunmugam, "Evaluation of straightness and flatness error using computational geometric techniques," *Computer-Aided Design*, vol. 31, no. 13, pp. 829–843, 1999.

[46] ——, "Evaluation of circularity from coordinate and form data using computational geometric techniques," *Precision Engineering*, vol. 24, no. 3, pp. 251–263, 2000.

[47] T. Santner, B. Williams, and W. Notz, *The Design and Analysis of Computer Experiments*. New York: Springer, 2003.

[48] O. Schabenberger and C. Gotway, *Statistical Methods for Spatial Data Analysis*. Boca Raton, Florida: Chapman & Hall/CRC, 2005.

[49] T. Shen, J. Huang, and C. Menq, "Multiple-sensor integration for rapid and high-precision coordinatemetrology," *IEEE/ASME Transactions on Mechatronics*, vol. 5, no. 2, pp. 110–121, 2000.

[50] M. Shunmugam, "New approach for evaluating form errors of engineering surfaces," *Computer-Aided Design*, vol. 19, no. 7, pp. 368–374, 1987.

[51] ——, "Criteria for computer-aided form evaluation," *Journal of Engineering for Industry*, vol. 113, no. 2, pp. 233–238, 1991.

[52] K. Stout, E. Davis, and P. Sullivan, *Atlas of Machined Surfaces*. London: Chapman & Hall, 1990.

[53] D. Suen and C. Chang, "Application of neural network interval regression method for minimum zone straightness and flatness," *Precision Engineering*, vol. 20, no. 3, pp. 196–207, 1997.

[54] J. Ton and A. Jain, "Registering landsat images by point matching," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 27, no. 5, pp. 642–651, 1989.

[55] M. D. Traband, M. and M. Chandra, "A statistical approach to tolerance evaluation for circles and cynlinders," *IIE Transactions*, vol. 36, pp. 777–785, 2004.

[56] T. Weber, S. Motavalli, B. Fallahi, and S. Cheraghi, "A unified approach to form error evaluation," *Precision Engineering*, vol. 26, no. 3, pp. 269–278, 2002.

[57] A. Weckenmann, M. Heinrichowski, and H. Mordhorst, "Design of gauges and multipoint measuring systems using coordinate-measuring-machine data and computer simulation," *Precision Engineering*, vol. 13, no. 3, pp. 203–207, 1991.

[58] C. Wikle and L. Berliner, "Combining information across spatial scales," *Technometrics*, vol. 47, no. 1, pp. 80–91, 2005.

[59] R. Wolfinger, R. Tobias, and J. Sall, "Computing Gaussian likelihoods and their derivatives for general linear mixed models," *SIAM Journal of Scientific Computing*, vol. 15, pp. 1294–1310, 1994.

[60] T. Woo and C. Hsieh, "Discussion: Statistical issues in geometric feature inspection using coordinate measuring machines," *Technometrics*, vol. 39, no. 1, pp. 20–22, 1997.

[61] B. Yang and C. Menq, "Compensation for form error of end-milled sculptured surfaces using discrete measurement data," *International Journal of Machine*

Tools & Manufacture, vol. 33, pp. 725–740, 1993.

[62] T. Yang and J. Jackman, "Form error estimation using spatial statistics," *Journal of Manufacturing Science and Engineering*, vol. 122, pp. 262–272, 2000.

[63] K.-M. Yeh, J. Ni, and S. Hu, "Adaptive sampling and identification of feature deformation for tolerance evaluation using coordinate measuring machines," University of Michigan-Ann Arbor, S. M. Wu Manufacturing Research Laboratory, Dept. of Mechanical Engineering and Applied Mechanics, Ann Arbor, MI, Tech. Rep., 1994.

[64] D. H. Zhu, X. and M. Wang, "Form error evaluation: An iterative reweighted least squares algorithm," *Transactions of the ASME, Journal of Manufacturing Science and Engineering*, vol. 126, pp. 535–541, 2004.

[65] D. Zimmerman and D. Harville, "A random field approach to the analysis of field-plot experiments and other spatial experiments," *Biometrics*, vol. 47, no. 1, pp. 223–239, 1991.

[66] B. Zitová and J. Flusser, "Image registration methods: A survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977–1000, 2003.

APPENDIX A

CONDITIONAL DISTRIBUTION THEOREM FOR MULTIVARIATE NORMAL DISTRIBUTION

Assume an n-dimensional random vector

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}$$

has a normal distribution $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix},$$

$\mathbf{y}_1$ and $\mathbf{y}_2$ are two subvectors of respective dimensions p and q with $p + q = n$. Note that $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}^T$, and $\boldsymbol{\Sigma}_{21} = \boldsymbol{\Sigma}_{12}^T$.

Then the distribution of $\mathbf{y}_1$ conditional on $\mathbf{y}_2 = \mathbf{a}$ is multivariate normal, $(\mathbf{y}_1 | \mathbf{y}_2 = \mathbf{a}) \sim N(\bar{\boldsymbol{\mu}}, \overline{\boldsymbol{\Sigma}})$ where the mean

$$\bar{\boldsymbol{\mu}} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\left(\mathbf{a} - \boldsymbol{\mu}_2\right),$$

and the covariance matrix

$$\overline{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}.$$

APPENDIX B

MARKOV CHAIN MONTE CARLO

Markov chain Monte Carlo (MCMC) methods are a class of algorithms for sampling from probability distributions. They construct a Markov chain that has the desired distribution, for example, a Bayesian posterior distribution, as its stationary distribution. The samples are drawn sequentially, with the distribution of the sampled draws depending only on the last value drawn. Hence, the draws form a Markov chain. (As defined in probability theory, a Markov chain is a sequence of random variables $\boldsymbol{\theta}^1, \boldsymbol{\theta}^2, \ldots$, for which, for any t, the distribution of $\boldsymbol{\theta}^t$ depends only on the most recent value, $\boldsymbol{\theta}^{t-1}$.) The states of the chain after a large number of steps are then used as a sample from the desired distribution.

In Bayesian statistics, MCMC is used when it is not possible or computationally inefficient to sample $\boldsymbol{\theta}$ directly from the posterior distribution $p(\boldsymbol{\theta}|\mathbf{y})$. For example, the posterior predictive distribution and the posterior distribution in (4.13) involve a high-dimensional integral and can not be derived analytically. Instead, we sample iteratively in a way such that at each step we expect to draw from a distribution closer and closer to $p(\boldsymbol{\theta}|\mathbf{y})$. For a wide class of Bayesian problems including posterior distributions from many hierarchical models, MCMC appears to be the easier way to get reliable results. A good reference on using MCMC for Bayesian posterior computations is [22]. MCMC also has many applications outside Bayesian statistics. For example, Reference [44] developed a MCMC-based algorithm for optimization.

The key to MCMC is to create a Markov chain whose stationary distribution is the desired $p(\boldsymbol{\theta}|\mathbf{y})$ and to run the simulation long enough such that the distribution of the current draws is close enough to the stationary distribution. For any specific

$p(\boldsymbol{\theta}|\mathbf{y})$, various MCMC algorithms with the desired property can be constructed. Here we introduce some basic methods: Gibbs sampler, Metropolis-hastings algorithm and slice sampling.

1. Gibbs sampler

Gibbs sampler is also called *alternative conditional sampling*. It is defined in terms of subvectors of $\boldsymbol{\theta}$. Suppose the parameter vector $\boldsymbol{\theta}$ has been divided into d subvectors, $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_d)$. Each iteration of Gibbs sampler cycles through the subvectors of $\boldsymbol{\theta}$, drawing each subvector conditional on the values of all the others: That is, for $j = 1, 2, \ldots, d$, sample from

$$p(\boldsymbol{\theta}_j|\boldsymbol{\theta}_{-j}^{t-1}, \mathbf{y}),$$

where $\boldsymbol{\theta}_{-j}^{t-1}$ represents all the components of $\boldsymbol{\theta}$, except for $\boldsymbol{\theta}_j$, at their current values.

Many problems involve standard statistical models, for example, the linkage model in (4.7) is essentially a linear regression model. For these problems, it is possible to analytically derive the conditional distributions of some parameters given other parameters and sample from them directly (as shown in Section IV.E.1). Under these cases, we will typically prefer Gibbs sampler since it provides easy and fast computations.

More often, however, we can not analytically derive the conditional distributions required by Gibbs sampler. Under those situations, we turn to more general MCMC methods, like Metropolis-Hastings algorithms or slice sampling. Gibbs sampler is actually a special case of Metropolis-Hastings algorithms.

2. Metropolis-Hastings algorithm

The Metropolis-Hastings algorithm adopts a random work and uses an acceptance rule to converge to the specified target distribution. The procedures proceed as follows:

(i). Draw a starting point $\boldsymbol{\theta}^0$, for which $p(\boldsymbol{\theta}^0|\mathbf{y}) > 0$, from a starting distribution $p_0(\boldsymbol{\theta})$. The starting distribution might be based on an approximation of the target distribution, or we may simply choose starting values dispersed around crude approximates.

(ii). For t=1,2,...

(a). Sample a proposal $\boldsymbol{\theta}^*$ from a jumping distribution (also called a proposal distribution) at time t, $J_t(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{t-1})$.

(b). Calculate the ratio of the densities,

$$r = \frac{p(\boldsymbol{\theta}^*|\mathbf{y})/J_t(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{t-1})}{p(\boldsymbol{\theta}^{t-1}|\mathbf{y})/J_t(\boldsymbol{\theta}^{t-1}|\boldsymbol{\theta}^*)}. \tag{B.1}$$

(c). Set

$$\boldsymbol{\theta}^t = \begin{cases} \boldsymbol{\theta}^* & \text{with probability } \min(r, 1), \\ \boldsymbol{\theta}^{t-1} & \text{otherwise.} \end{cases} \tag{B.2}$$

Given the current value $\boldsymbol{\theta}^{t-1}$, the transition distribution $T_t(\boldsymbol{\theta}^t|\boldsymbol{\theta}^{t-1})$ of the Markov chain is thus a mixture of a point mass at $\boldsymbol{\theta}^t = \boldsymbol{\theta}^{t-1}$ and a weighted version of the jumping distribution $J_t(\boldsymbol{\theta}^t|\boldsymbol{\theta}^{t-1})$ that adjusts for the acceptance rate. A commonly used jumping distribution is a normal distribution, for example, $N(\boldsymbol{\theta}^{t-1}, \sigma^2\mathbf{I})$. The value of σ^2 is tuned for a reasonable acceptance rate.

3. Slice sampling

Slice sampling is based on the principle that one can sample from a distribution by sampling uniformly from the region under the plot of its density function. Suppose we would like to sample from a distribution for a variable $\boldsymbol{\theta}$, whose density is proportional to some function $f(\boldsymbol{\theta})$. We can do this by sampling uniformly from the region that lies under the plot of $f(\boldsymbol{\theta})$. This idea can be formalized by introducing an auxiliary real variable U, and defining a join distribution over $\boldsymbol{\theta}$ and U that is uniform over the region below the curve defined by $f(\boldsymbol{\theta})$, $D = \{(\boldsymbol{\theta}, U) : 0 < U < f(\boldsymbol{\theta})\}$. In other words, the joint density for $(\boldsymbol{\theta}, U)$ is

$$p(\boldsymbol{\theta}, U) = \begin{cases} 1/Z & \text{if } 0 < U < f(\boldsymbol{\theta}), \\ 0 & \text{otherwise,} \end{cases}$$

where $Z = \int f(\boldsymbol{\theta}) d\boldsymbol{\theta}$. If we let $f(\boldsymbol{\theta})$ be the product of the likelihood and the prior density function, i.e., $f(\boldsymbol{\theta}) = L(\boldsymbol{\theta}, \mathbf{y})\pi(\boldsymbol{\theta})$, the marginal density for $L(\boldsymbol{\theta}, \mathbf{y})$ is then the posterior distribution of $\boldsymbol{\theta}$:

$$\int_0^{f(\boldsymbol{\theta})} (1/Z) dU = f(\boldsymbol{\theta})/Z = \frac{L(\boldsymbol{\theta}, \mathbf{y})\pi(\boldsymbol{\theta})}{\int L(\boldsymbol{\theta}, \mathbf{y})\pi(\boldsymbol{\theta}) d\boldsymbol{\theta}} = p(\boldsymbol{\theta}|\mathbf{y}).$$

Generating independent samples drawn uniformly from D may not be easy. So we might instead construct a Markov chain that will converge to this uniform distribution.

Gibbs sampler is one possibility: We sample alternately from (a) the conditional distribution for U given the current $\mathbf{y}$, which is uniform over the interval $(0, f(\boldsymbol{\theta}))$, and from (b) the conditional distribution for $\boldsymbol{\theta}$ given the current U, which is uniform over the region $D = \{(\boldsymbol{\theta}, U) : 0 < U < f(\boldsymbol{\theta})\}$, and it is called the "slice" defined by $\boldsymbol{\theta}$.

Here we use a single-variable slice sampling method to illustrate the procedures

to update θ in each iteration,

(i). Draw a real value, U, uniformly from $(0, f(\theta^{t-1}))$, thereby defining a horizontal "slice": $S = \{\theta : U < f(\theta^{t-1})\}$. Note that θ^{t-1} is always in S.

(ii). Draw θ^* from $\pi(\theta)$.

(iii). If θ^* is not in the slice S and is bigger than θ^{t-1}, then draw from $\pi(\theta)I(\theta < \theta^*)$. If θ^* is not in the slice S and is smaller than θ^{t-1}, then draw from $\pi(\theta)I(\theta > \theta^*)$. Repeat until a point in the slice is found.

The multi-variate case works either by applying the single-variable slice sampling procedures described above for each parameter in turn or by shrinking hyper-rectangles. One benefit of slice sampling is that it requires little or no tuning. Reference [39] provided more details on slice sampling.

This dissertation combines Gibbs sampler and slice sampling for approximating the posterior distribution and the posterior predictive distribution in (4.13). Section IV.E.1 derives the conditional distributions needed to implement Gibbs sampler for the parameters involved in the neighborhood linkage model. For the parameters involved in the low-resolution model, this dissertation adopts the single-variable slice sampling procedure and cycles through the parameters, sampling each parameter conditioned on the values of all the others.

APPENDIX C

BAYESIAN MODEL AVERAGING

Bayesian model averaging provides a way accounting for model uncertainty. If Δ is the quantity of interest, such as a future observable, then its posterior distribution given the data D is

$$p(\Delta|M_k) = \sum_{k=1}^{K} p(\Delta|M_k, D)p(M_k|D), \tag{C.1}$$

where $M_1, \ldots, M_K$ are the models considered. This is an average of the posterior distributions under each of the models considered, weighted by their posterior model probabilities.

The posterior probability for Model M_k is:

$$p(M_k|D) = \frac{p(D|M_k)p(M_k)}{\sum_{l=1}^{K} p(D|M_l)p(M_l)}, \tag{C.2}$$

where

$$p(D|M_k) = \int p(D|\boldsymbol{\theta}_k, M_k)p(\boldsymbol{\theta}_k|M_k)d\boldsymbol{\theta}_k,$$

$\boldsymbol{\theta}_k$ is the vector of parameters of Model M_k (e.g., for a regression model M_k, $\boldsymbol{\theta}_k = (\boldsymbol{\beta}, \sigma^2)$), $p(\boldsymbol{\theta}_k|M_k)$ is the prior probability of $\boldsymbol{\theta}_k$ under Model M_k, $p(D|\boldsymbol{\theta}_k, M_k)$ is the likelihood, and $p(M_k)$ is the prior probability that M_k is the true model.

By averaging over different competing models, Bayesian model averaging incorporates model uncertainty into the conclusions about parameters and predictions. Bayesian model averaging has been applied successfully to many statistical models including linear regressions, generalized linear models, and discrete graphical models. In all these cases, it improves predictive performances.

VITA

Ms. Haifeng Xia received her B.S. degree in international enterprise management from Tianjin University, China, in 2000 and her M.S. degree in management science and engineering from the same university in 2002. She received her Ph.D. in industrial and systems engineering in December, 2008 from Texas A&M University, College Station.

Ms. Xia's research interests are in quality engineering and applied statistics. Her doctoral research focused on multi-scale/multi-resolution modeling and analysis as well as its applications in dimensional quality control, nano-technology and remote sensing. She received the 2007 "Quality, Statistics and Reliability" Best Student Paper Award sponsored by INFORMS. Ms. Xia is a student member of INFORMS (the Institute for Operations Research and the Management Sciences), ASA (American Statistical Association) and IIE (Institute of Industrial Engineers).

Ms. Xia may be reached at 238 Zachry, MS 3131, Texas A&M University, College Station, TX 77843. Her email address is heidi.xia08@gmail.com.